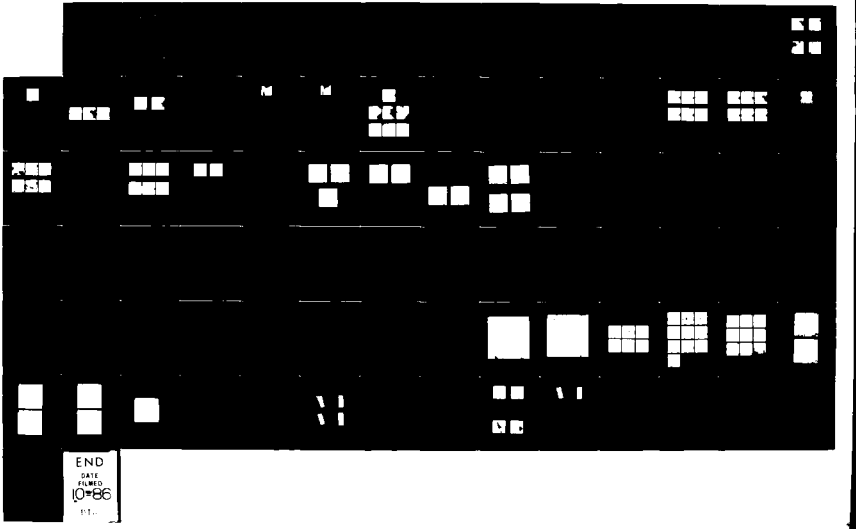
AD-A170 884 INTEGRATION OF ARTIFICIAL INTELLIGENCE CONCEPTS INTO THE METHODS FOR EXTR..(U) FORSCHUNGSINSTITUT FUER INFORMATIONSVERARBEITUNG KARLSRUHE (G.. R NEU ET AL. 1/1

UNCLASSIFIED MAR 86 FIL-0425 DAJA45-84-C-0014 F/G 14/5 NL

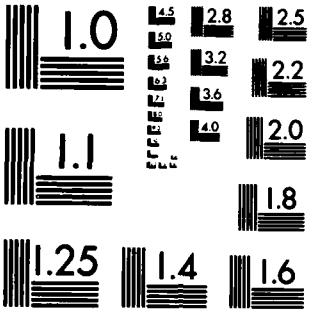


END

DATE FILMED

10-86

DTIC



ETL-0425

Integration of Artificial Intelligence Concepts into the Methods for Extracting Line Objects from Monochromatic Aerial Imagery

Final Technical Report

March 1986

EUROPEAN RESEARCH OFFICE
United States Army
London England

CONTRACT NUMBER DAJA 45-84-C-0014

FIM/FGAN, Eisenstockstr. 12
D-7505 Ettlingen 6, W. Germany

U. S. ARMY ENGINEER TOPOGRAPHIC LABORATORIES
FORT BELVOIR, VIRGINIA 22060-5546

Integration of Artificial Intelligence Concepts into the Methods for Extracting Line Objects from Monochromatic Aerial Imagery

Final Technical Report
by
R. Neu, W. Heißler
H. Kazmierczak, M. Sties

March 1986

EUROPEAN RESEARCH OFFICE
United States Army
London England

CONTRACT NUMBER DAJA 45-84-C-0014

FIM/FGAN, Eisenstockstr. 12
D-7505 Ettlingen 6, W. Germany

Principal Investigator: Prof. Dr.-Ing. H. Kazmierczak

FIM Report No. 156

The Research reported in this document has been made possible through the support and sponsorship of the U.S. Government through its European Research Office of the U.S. Army. This report is intended only for the internal management use of the Contractor and the U.S. Government.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ETL-0425	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* INTEGRATION OF ARTIFICIAL INTELLIGENCE ASPECTS INTO THE METHODS FOR EXTRACTION OF LINE OBJECTS FROM AERIAL IMAGES		5. TYPE OF REPORT & PERIOD COVERED Final Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Prof. Ing. H. Kazmierczak		8. CONTRACT OR GRANT NUMBER(s) DAJA45-84-C-0014
9. PERFORMING ORGANIZATION NAME AND ADDRESS Forschungsinstitut fur Informations-verarbeitung und Mustererkennung Eisenstockstr.12, D-7505 Ettingen 6, W.GERMANY		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102A 1T161102BH57-01
11. CONTROLLING OFFICE NAME AND ADDRESS USARDSG-UK Box 65, FPO NY 09510		12. REPORT DATE March 1986 13. NUMBER OF PAGES 78
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)* U. S. ARMY ENGINEER TOPOGRAPHIC LABORATORIES FORT BELVOIR, VIRGINIA 22060-5546		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Automatic image processing; object extraction; artificial intelligence aspects; digital elevation data;

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

Procedures for automatic extraction of line shaped objects from aerial images have been improved and completed. A general model of road network has been used to complete road extraction from images. Digital elevation data has been used to guide the process of river and creek extraction from images. The methods have been implemented on a DEC VAX 11/780. The functions are described in detail. Test results and assessment are included in the report.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

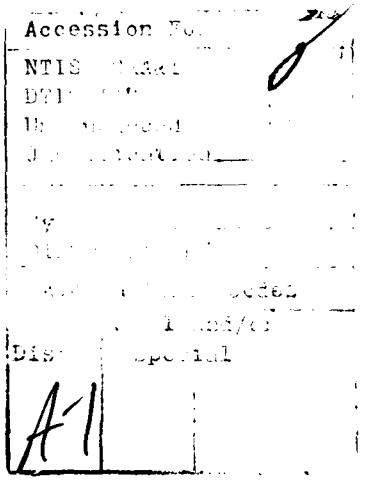


Table of Contents

1. Statement of the Problem

The development of methods for the extraction of line objects from aerial imagery has been pursued for several years at FIM. Results have been presented and discussed in earlier final technical reports /1,2/. Comparing e.g. the total network of roads which can be seen visually in an aerial image (see figure 4.15) with the result of a fully automatic line extraction procedure /2, figure 5.28 on page 85/ had led to the conclusion that the computer result was still lacking completeness due to some missing road segments which could not be detected automatically. To enhance the performance of the extraction procedure, the improvement of the signal processing algorithms did not seem as promising as the integration of artificial intelligence aspects.

Three different aspects of artificial intelligence were taken into consideration:

i the general model of the network of man made objects such as roads, highways, railroads, etc. suggests that only very few of these objects will end in dead ends. In fact, every dead end of an automatic extraction result, where a line continuation was not accepted or could not even be detected, remains a candidate for a completion process.

ii special "a priori" knowledge about the relations between line objects and surrounding other objects can help to enhance the performance of the extraction procedure. For example, a road may often be parallel or perpendicular to contours of surrounding objects such as agricultural fields or forests.

iii the extraction of the course of rivers and creeks from aerial images would be improved substantially by the integration of elevation data into the extraction process, because we could then exploit the fact, that rivers are flowing in valleys and, additionally, that rivers are mostly located in the lowest part of the valleys.

Hence, we had to work on three topics which can be summarized as follows:

i analyse a given line extraction result to detect gaps in the line network. At the location of these gaps, a verification procedure has to be performed in the aerial image.

ii analyse a given line extraction result to detect gaps in the line network. At the location of these gaps, a detailed analysis of surrounding object areas has to be performed in order to deduce the existence and possible orientations of missing line segments. These predictions again have to be verified in the aerial image.

iii include a matrix of elevation data into the procedure of coincident line extraction from two corresponding input data sets: iterative application of an appropriate sequence of extraction, prediction, and verification.

To work on these problems, we could use the same aerial images No. 54 and No. 56 as test data, as we already had done in previous projects. This allows direct comparison with earlier results. In addition, USAETL provided one example of a digital elevation matrix which had been produced by stereo correlation of the same two aerial images. The elevation matrix consisted of 395 x 390 elevation data and was accompanied by a complete description to enable registration to the aerial images.

Our solutions to the problems described are explained and discussed in the following sections of this report. Reference is made to eight progress reports which have been produced and distributed during work on the contract.

Although it was the aim of this project to produce a complete software implementation of the extraction procedures, it should be understood that we were not able to optimize neither computer run time nor computer memory requirements; we also did not adapt the software implementation to any of the various special purpose computing systems to optimize operations.

2. Basic Methods

Basic methods to be modified and extended for proper application in this research project have been developed and tested during earlier projects/1,2/. This refers to:

i the procedure for starting point search

ii the local extraction method (LEM)

iii the regional extraction method (REM)

The procedure for starting point search analyses sample lines resp. sample columns of the image to detect short segments of line objects, the features of which (width, contrast, straightness, etc.) comply with predefined model segments. The local extraction method works on an onedimensional, semicircular sample line taken from the image at the very neighbourhood of the actual processing position; the grey values of the sample line are analysed to detect a next cross section of the line object which corresponds to the cross section of the actual processing position. The regional extraction method works on an "area of interest", the location of which is predicted from the actual processing position; the grey values of the area of interest are analysed to detect several collinearly located cross sections of the line object which can be accepted as a continuing segment of this object.

A combined application of these three basic methods automatically produces line extraction results from aerial imagery. Considerable sophistication has been integrated into the extraction methods to tolerate local distortions and noise. However, the extraction will stop in all cases, where the decision in favour of the line continuation is not reliable. At these locations, a variety of processing steps could be considered to eliminate extraction stops including but not confined to the following:

- changing parameter values

- changing processing locations

- changing acceptance thresholds.

The decisions, however, would still be based on local grey level evaluation and would not be more reliable due to lack of "global" context. Hence, we decided to follow other ideas which will be explained in the following.

3. Improvement of River Extraction

3.1 Preparations for River Extraction

3.1.1 Basic Considerations

To integrate more artificial intelligence aspects into the methods for the extraction of line objects from aerial images, digital elevation data is used as context. The aim of the extension is to automate the extraction of rivers and creeks. Objectives of this part of the project are:

(1) Analysis of the digital elevation data and integration into the software system.

(2) Modifications of the extraction procedures to work on elevation data.

(3) Modifications of the verification procedure to work on rivers and creeks.

(4) Extensive test, assessment and documentation.

The general proceeding in extracting rivers and creeks is as follows:

- Extract the valleys from the elevation data.
- Transform the coordinates of the extracted valleys to the image data system.
- Use the transformed valley coordinates to predict rivers and creeks.
- Verify these predictions by analysis of the image data.
- Perform a continuation extraction of rivers and creeks from the image data.
- Transform the continuation extraction result back to the elevation data system.
- Verify that the result does not contradict the elevation data.

This proceeding agrees with the concept of coincident extraction and therefore permits the use of a number of procedures already existing.

3.1.2 Assessment of the Existing Program System

Some important facts have to be considered when using the already existing methods for the extraction of rivers and creeks from the aerial image data and valleys from the elevation data.

We have used image No. 56 as test data. With respect to the extraction of rivers and creeks from the image data these are:

- The width of a river may be less constant than the width of a road.
- The course of a river may be full of bends.

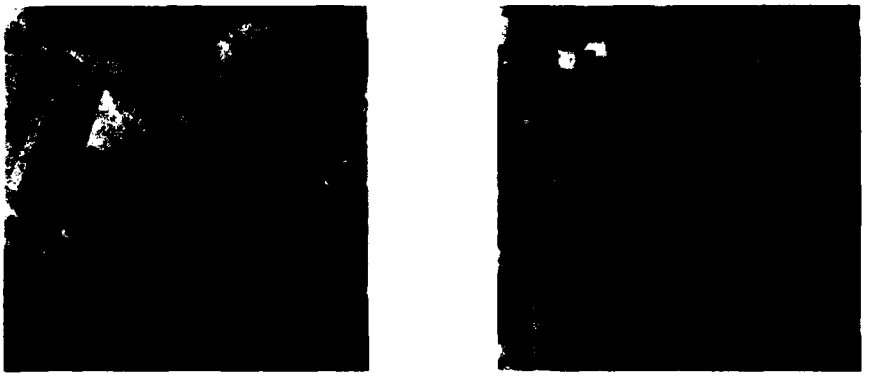
With respect to the extraction of valleys from the elevation data these are:

- Valleys may be very wide.
- The width of a valley may vary considerably.

An examination of the valley width in the elevation matrix and the river width in the image matrix showed that we had to reduce the elevation matrix by a factor of 2 (figures 3.1a and b) and the image matrix by a factor of 4 to get the object width expected by the procedures already available. Figures 3.2a and b display two sections of different resolutions from the aerial image. Figure 3.2a displays a section used for the extraction of roads and figure 3.2b displays a section used for the extraction of the river. The average width of the main valley in the reduced elevation matrix of 194 x 194 pixels is 5 pixels and the average width of the river in the reduced image matrix of 1024 x 1024 pixels lies between 2 to 4 pixels.

(a) (b)

Fig. 3.1: Different resolutions of the elevation data
(a) 388 x 388 pixels elevation matrix
(b) 194 x 194 pixels elevation matrix



(a) (b)

Fig. 3.2: Different resolutions of the image data
(a) Part of the aerial image for road extraction
(b) Part of the aerial image for river extraction

To assess the efficiency of the old methods with respect to the extraction of the river in the aerial image without the use of the elevation matrix, we have started the extraction process for several different sets of parameter values. One extraction result is displayed in figure 3.3. The result indicates that the use of additional information, e.g. in the form of elevation data, is necessary to guide the extraction process.

Fig. 3.3: Automatic river extraction performed by the already existing system

Thus, modifications and extensions of some of the main programs were necessary, to implement the cooperation between elevation and image data processing. The main programs primarly affected were the program for the starting-point search, the program for the centering of the starting-points, the line verification program and the line extraction program, which have been developed during the last contracts.

3.1.3 Examination of the Elevation Data

For the extraction of natural line objects like rivers and creeks an elevation matrix has been provided on tape by USAETL. The elevation data cover nearly the whole area mapped by the aerial image shown in figure 3.4a at a reduced scale. Since our extraction methods require a byte matrix, we had to rescale the elevation data into the range of 0 to 255. The result of this transformation can be used like a graylevel image. The transformation was performed by a simple shift, because the difference between the minimal and maximal height of the imaged area amounted to 253 meters (figure 3.4b). Thus, in our rescaled elevation matrix the graylevel 0 represents the height of 275 meters and the graylevel 253 represents the height of 528 meters.

3.1.3.1 Registration between Image Data and Elevation Data

A registration between the image data and the elevation data was performed by means of 22 control points defined by USAETL. These control points are distributed all over the elevation data. A number of drawings of the corresponding locations in the aerial image enabled us to determine their approximate coordinates in the image coordinate system. To check the accuracy of these definitions, the elevation data (figure 3.4b) were mapped by the polynomial method, implemented during the last contract, to the image data system. The transformation result is shown in figure 3.4c.

(a) (b) (c)

Fig. 3.4: (a) Aerial image No.56
(b) Magnified elevation matrix (512 x 512 pixels)
(c) Mapped, magnified elevation matrix (512 x 512 pixels)

3.1.3.2 Manual Determination of the Course of the River

To get a first impression of the approximate location of the river in the elevation matrix, the course of the river was manually determined in the aerial image matrix and transformed to the coordinate system of the elevation matrix by the polynomial method. Figures 3.5a and b show the results. Though the manual determination of the course of the river was performed in segments of the original image matrix, it was not possible to deter-

mine the course of the river exactly in the urban area and in the narrow valley in the right top corner of the image (figures 3.5a and b). Besides the main stream only very short branches of the river have been detected. The location of the lake is also displayed. Though the elevation matrix contains several small valleys, a water course is only in the main valley.

(a) (b)

Fig. 3.5: Manually determined water bodies
(a) Water bodies in the aerial image
(b) Water bodies transformed to the elevation matrix

3.1.3.3 Assessment of the Quality of the Elevation Data

Starting from the manually determined and transformed course of the river, some tests were performed to assess the reliability of the elevation data. For this purpose some sample histograms and the corresponding graylevel windows were evaluated. Two examples are displayed in figures 3.6 and 3.7. A coarse quantification of grayvalue levels of the elevation matrix indicates that the elevation values oscillate within a relative short distance. Figure 3.8 shows a 4 bit quantification of grayvalue levels, Figures 3.9a-c display segments from this representation containing the main valley and demonstrate that the elevation data along the valley are not descending monotonously. Figures 3.9d-f show the corresponding segments from the aerial image.

The tests have proved:

- At every location the valley is clearly detectable.
- The flow direction of the river cannot be calculated unambiguously from the elevation data, which is probably due to the narrowness of the valley and the hiddenness of slopes. At some places the elevation values correspond to ground levels, at other places they correspond to the levels of tree tops. The same problem occurs in the urban area.
- The elevation data of the lake are unreliable, as could be expected from missing correlation features.

(a)

54	54	54	43	38	37	39	40	41	43	44	48	53	56	60	59	61	63	64	65	66	66	66	65	64	64
53	52	45	41	38	38	39	40	42	45	49	59	60	62	57	56	60	62	64	66	68	67	66	65	62	64
52	48	43	40	38	39	41	44	46	52	52	56	70	74	58	57	61	63	63	64	66	68	63	63	63	63
51	46	41	38	38	40	42	44	49	52	57	61	72	57	57	55	62	67	62	64	65	65	66	67	64	61
48	45	39	37	38	40	43	47	52	56	60	63	55	52	54	54	54	60	63	63	64	62	62	61	41	59
49	47	34	37	37	41	44	48	55	51	51	54	56	53	54	56	54	56	61	62	62	61	61	64	59	59
51	51	32	37	37	38	42	44	46	49	51	51	52	53	55	56	55	56	59	61	61	61	62	62	61	49
53	51	45	37	37	38	40	43	46	47	51	53	53	55	55	55	55	56	59	59	60	61	61	61	64	63
56	51	47	40	38	38	38	40	42	46	47	53	52	54	50	53	50	49	49	49	57	60	61	63	63	64
58	54	50	46	42	40	39	39	40	45	45	52	52	50	50	52	50	55	55	52	55	61	60	64	63	64
59	57	52	48	43	41	39	39	40	43	46	49	54	54	55	51	52	53	51	51	52	56	61	63	62	63
58	58	55	50	46	43	40	39	39	40	47	49	53	54	53	52	55	56	53	52	53	54	60	63	61	62
58	58	58	54	48	44	41	39	39	39	46	45	47	50	49	49	53	52	53	50	54	54	55	57	60	61
57	58	58	56	52	49	43	40	39	39	39	43	50	53	49	50	56	56	55	49	55	55	55	55	60	60
53	55	56	55	51	50	49	43	39	38	37	44	46	47	44	50	50	50	54	44	56	55	55	51	53	56
52	52	52	53	52	51	50	45	39	37	37	36	45	44	52	52	51	51	52	53	46	50	56	54	55	54
51	52	52	52	51	51	51	47	39	35	39	44	38	41	44	46	49	48	48	47	54	48	50	48	52	52
52	51	51	51	51	51	50	47	44	40	36	28	38	38	39	43	43	44	45	46	50	50	49	52	52	52
52	51	51	51	51	51	51	48	45	43	42	35	35	42	42	29	41	42	43	45	44	46	47	51	52	53
52	53	51	50	50	50	50	48	45	45	43	39	36	40	41	43	40	42	43	45	45	48	45	48	52	54
53	53	51	50	50	51	51	50	45	45	45	43	38	39	37	37	38	41	44	45	46	46	46	47	49	54
60	57	52	50	50	50	51	46	46	45	44	43	39	39	40	38	37	41	43	45	46	47	47	48	48	48
65	57	55	51	45	43	45	46	46	44	43	43	45	42	41	38	37	38	42	43	45	46	47	48	49	50
60	59	56	55	55	46	40	41	45	47	45	47	46	42	42	40	39	39	43	41	45	46	48	48	48	49
66	59	56	52	50	48	45	45	45	45	47	48	46	43	43	42	39	39	39	39	42	45	43	43	47	48
59	59	57	55	53	51	51	49	46	56	41	45	44	44	43	41	40	39	40	43	34	45	39	42	47	48
56	56	57	55	52	52	52	52	57	50	56	47	47	47	46	44	45	41	40	39	43	44	39	41	45	47
57	56	56	55	53	52	53	52	55	51	55	52	54	44	44	45	45	50	37	37	40	40	39	36	41	42

(b)

Fig. 3.6: Different representations of elevation data
(a) A graylevel window with its graylevel histogram
(b) A graylevel matrix of the central part of the window

(a)

47	55	55	55	54	54	54	52	52	50	52	51	49	49	49	49	48	47	42	42	43	43	42	41	41	41
57	62	58	53	53	53	52	48	48	47	49	48	47	48	51	51	44	48	41	45	37	40	46	44	36	36
46	48	49	51	51	49	48	45	45	47	48	45	46	49	46	48	51	39	40	41	41	41	40	40	37	39
48	47	47	45	50	49	45	43	43	42	42	42	42	47	49	51	42	41	41	40	40	38	36	36	37	37
45	46	46	45	45	45	45	42	42	42	41	41	42	41	41	42	44	42	42	40	38	38	36	36	36	36
39	43	44	45	44	44	45	42	42	41	41	40	36	41	38	41	43	41	41	39	37	37	36	36	36	36
39	41	42	42	41	37	39	40	40	40	38	36	36	37	38	40	43	42	41	38	36	36	35	35	35	35
37	38	39	38	39	38	37	36	37	38	37	35	35	37	36	37	36	36	35	35	35	36	35	36	36	35
44	44	42	37	39	38	37	35	35	37	35	35	35	37	35	36	36	36	35	36	36	36	36	36	35	40
46	45	44	41	42	39	37	35	34	34	33	34	34	37	34	36	32	35	35	35	36	36	36	36	36	38
46	45	44	42	42	40	37	33	33	32	32	36	31	35	35	36	35	35	35	35	35	36	36	37	37	37
42	43	43	40	38	37	36	34	37	32	33	32	32	32	34	33	35	34	35	34	34	35	36	38	39	40
38	39	37	36	34	34	33	37	37	34	34	34	33	32	33	32	33	32	34	34	34	35	36	39	40	40
37	35	32	36	34	32	32	33	32	33	34	34	34	34	34	34	34	33	34	35	35	35	36	41	42	42
38	32	32	32	33	33	32	32	32	33	34	34	34	35	34	34	34	34	35	35	36	35	40	40	42	42
34	33	33	33	33	33	33	33	33	34	34	35	36	36	35	35	36	35	35	35	37	37	41	42	43	44
34	35	34	35	34	34	34	34	34	35	33	35	37	36	37	38	38	36	36	36	38	40	41	42	42	43
37	37	36	36	36	36	36	36	38	45	36	38	39	38	40	40	40	40	40	38	38	36	41	42	45	45
43	39	38	37	37	38	40	42	39	36	36	39	41	42	42	42	42	41	39	37	37	37	40	44	46	44
45	43	42	40	39	40	41	42	43	43	38	39	40	41	41	42	41	41	37	37	37	38	41	43	42	43
57	48	45	49	43	41	42	42	42	43	39	38	40	40	40	41	40	39	36	35	36	41	43	44	43	43
61	51	49	48	45	43	43	43	43	42	43	38	37	37	36	38	37	37	36	36	37	41	42	43	42	42
62	58	53	50	46	43	43	43	42	41	43	38	39	38	37	37	36	36	36	35	40	40	41	42	46	42
63	57	50	49	46	44	44	43	43	40	46	40	38	39	40	39	39	37	36	34	38	39	39	40	49	42
62	58	46	43	45	44	44	43	41	53	47	42	41	41	42	41	41	39	37	35	37	39	39	40	48	43
64	58	45	44	44	45	46	44	48	43	50	42	42	41	41	42	42	40	37	36	35	38	38	39	41	43
64	56	47	47	44	46	49	49	49	47	50	46	42	42	42	42	42	41	42	41	36	37	37	39	41	42
65	53	45	50	44	49	48	46	43	43	44	43	42	42	42	45	43	42	42	40	39	37	38	39	39	41

(b)

Fig. 3.7: Different representations of elevation data
(a) A graylevel window with its graylevel histogram
(b) A graylevel matrix of the central part of the window

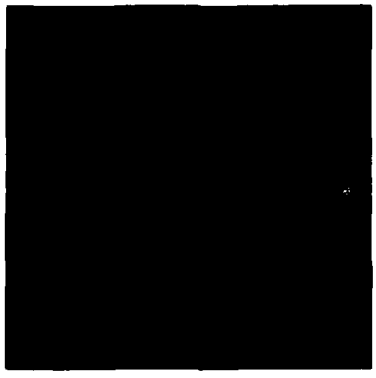


Fig. 3.8: 4 bit quantification of the elevation data

(a) (b) (c)

(d) (e) (f)

Fig. 3.9: (a)-(c) Segments of the elevation matrix and
(d)-(f) corresponding segments of the image matrix

3.2 The New Procedure for Automatic River Extraction

There are several possibilities to integrate the elevation data into the process of river and creek extraction. One promising approach is to use the cooperation of the extraction methods developed during the last contract. Two applications of this cooperation are conceivable:

(1) Beginning with the aerial image

- Search for starting-points in the image matrix.
  Transformation of the starting-points to the elevation data system.
  Valley extraction from the elevation data starting from the centered starting-points.
  Transformation of the extracted valleys back to the image data system.
  Verification of the extracted valleys in the image data by extraction of water courses.

- Search for starting-points in the image matrix and river extraction from the image data.
  Transformation of the river extraction result to the elevation data system.
  Verification of the river extraction result in the elevation data by valley extraction.
  Continued valley extraction from the elevation data.
  Transformation of the continued valley extraction result to the image data system and verification in the image data.

(2) Beginning with the elevation data

- Search for starting-points in the elevation matrix.
  Transformation of the starting-points to the image data system.
  Centering of the transformed starting-points in the image data system.
  River extraction from the image data starting from the ver-

ified starting-points.

- Search for starting-points in the elevation matrix.
  Valley extraction from the elevation data.
  Transformation of the extraction result to the image data system.
  Verification of the extraction result in the image data.
  Continuation extraction from the image data.

Due to complex situations and concealed parts of the river in the aerial image, it seems more promising to begin the extraction process with the elevation data than with the image data. Since the elevation matrix is smaller than the image matrix, this proceeding has the further advantage that the search for starting-points and the subsequent extraction process will be much quicker.

3.2.1 Search for Starting-points in the Elevation Matrix

The extraction process begins with the search for starting-points. This is done much quicker in the elevation matrix than in the image matrix, because the original elevation matrix is 395 x 390 pixels large and the reduced elevation matrix is 194 x 194 pixels large. It is true that when we extract a valley from the elevation data, we do not know whether a river is flowing there. On the other hand the elevation data are much easier to interprete than the image data. For example, it may be sometimes impossible for the extraction procedure to distinguish between an hedge or the edge of a forest and a river or a creek in the image data, but searching for valleys in the elevation data most probably leads to a very small number of misinterpretations.

Some modifications in connection with the starting-point search were necessary:

- The procedure for the starting-point search was integrated as a subprogram into the old program system for the extraction of

line objects. We had to develope a new implementation of the procedure as a stand alone program, to perform a search over the whole pattern without starting the extraction process.

- Since the width of a valley may vary significantly, some parameter adaptations were necessary. The sets of parameter values chosen for tests were not very restrictive.
- A new parameter CR has been defined to the effect, that the contrast requirement for a cross-section of a line object becomes a function of the graylevel of the pixel in the middle of the cross-section, thus accounting for the fact that usually valleys in the mountains have steeper slopes than valleys in the plain. A cross-section is accepted as a cross-section of a valley only if the following condition is satisfied:

$$((GR + (GM / 255) * CR) < |GM - GRM|) \quad \text{AND} \quad ((GR + (GM / 255) * CR) < |GM - GLM|) \qquad (3 - 1)$$

with
GR: basic contrast requirement
GM: graylevel of the pixel in the middle of the cross-section
GLM: graylevel of the left-most pixel of the cross-section
GRM: graylevel of the right-most pixel of the cross-section

If CR is greater than zero, then the value of the left sides of the inequalities increases with increasing value of GM. Thus, the values of the right sides of the inequalities have to increase too for the condition to be true. In other words: the difference in elevation required between the edge of a valley and its bottom increases with increasing elevation of the bottom.

The results of the experiments with the original starting-point search which are displayed in figures 3.10a-c were achieved by using the 388 x 388 pixels elevation matrix and by limiting the object width to 5 to 24 pixels and the distance between the sample lines to 20 to 25 pixels.

The results displayed in figures 3.10d-f have been produced by means of the 194 x 194 pixels elevation matrix, an object width of 3 to 12 pixels and a distance between sample lines of 5 to 10 pixels.

In our experiments with the 388 x 388 pixels elevation matrix, the new parameter CR was set to 10. Another experiment was performed with the 194 x 194 pixels elevation matrix, where the parameter CR ranged between 20 and 30.

The modified program for the starting-point search was successfully applied to the 388 x 388 pixels elevation matrix as well as to the reduced elevation matrix. Results of a starting-point search with the distance between the sample lines ranging from 6 to 12 pixels and the width of the objects ranging from 3 to 25 pixels are displayed in figures 3.11a-c. The only difference between the results displayed in figures 3.10d-f and the results displayed in figures 3.11d-f is the difference in the value of CR, which was set to 30 (figure 3.11d) and to 20 (figures 3.11e and f) in the second case.

Both alternatives are useful for different purposes:

- If a river extraction from the aerial image is to follow the search for starting-points in the elevation matrix, we prefer the 388 x 388 pixels elevation matrix.

- If a valley extraction has to be performed from the elevation data, we prefer the 194 x 194 pixels elevation matrix for the starting-point search.

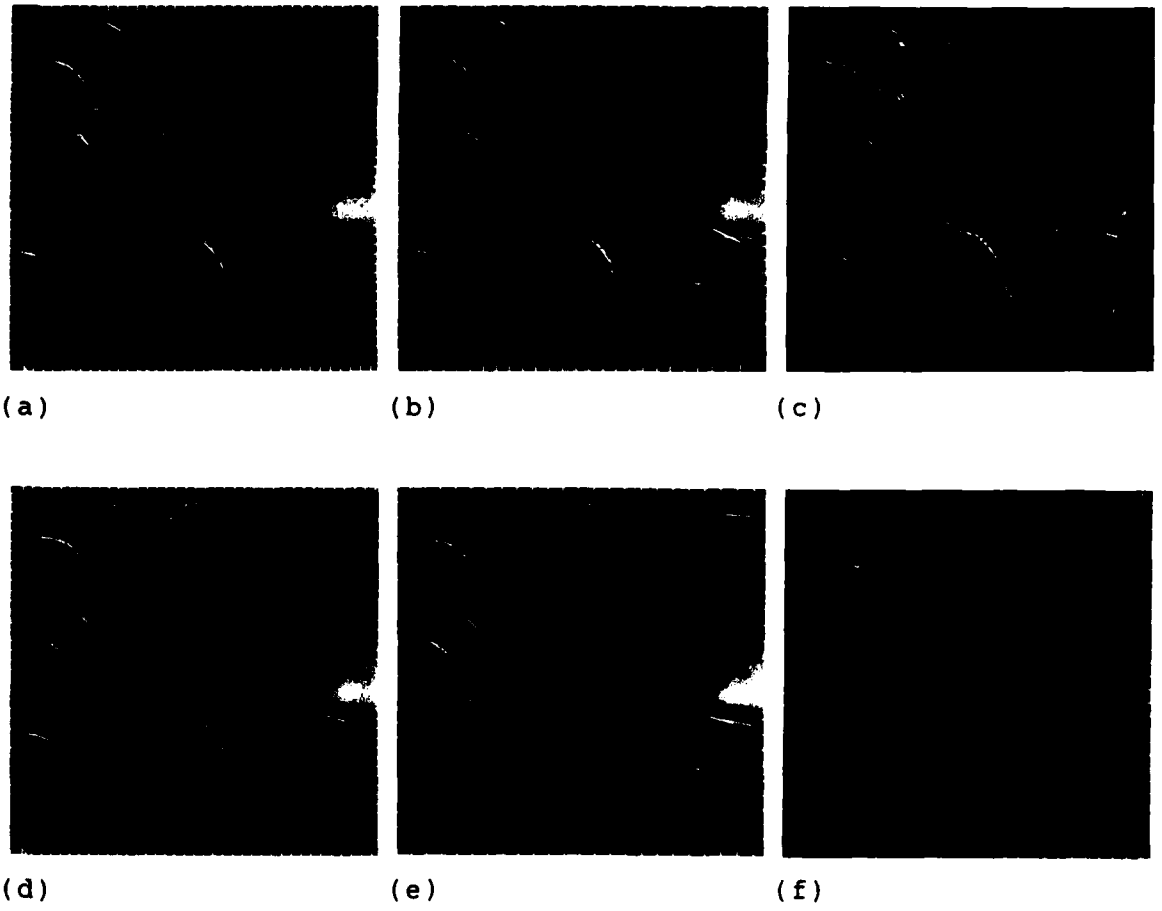


Fig. 3.10: Original starting-point search in the elevation matrix
(a)-(c): 388 x 388 pixels elevation matrix
(d)-(f): 194 x 194 pixels elevation matrix

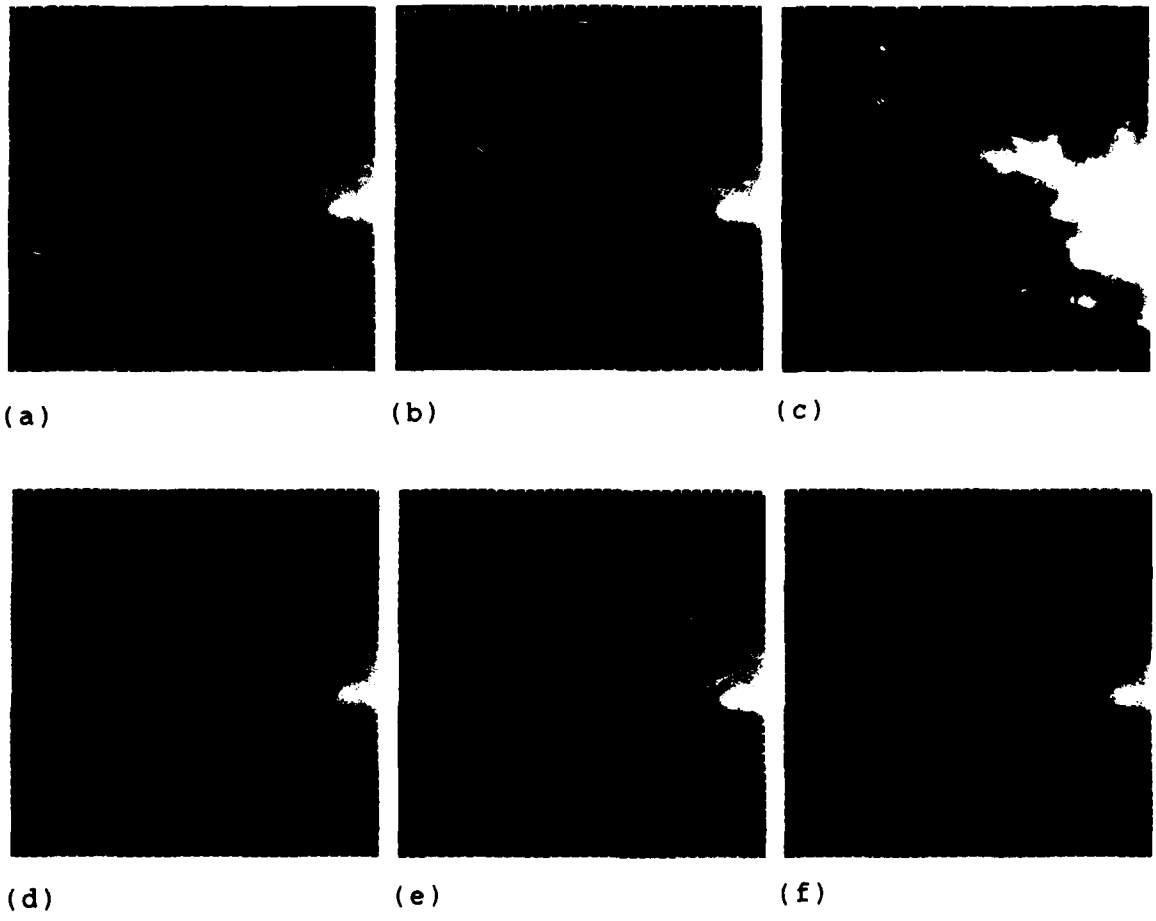


Fig. 3.11: Results of the modified starting-point search in the elevation matrix
(a)-(c) 388 x 388 pixels elevation matrix
(d)-(f) 194 x 194 pixels elevation matrix

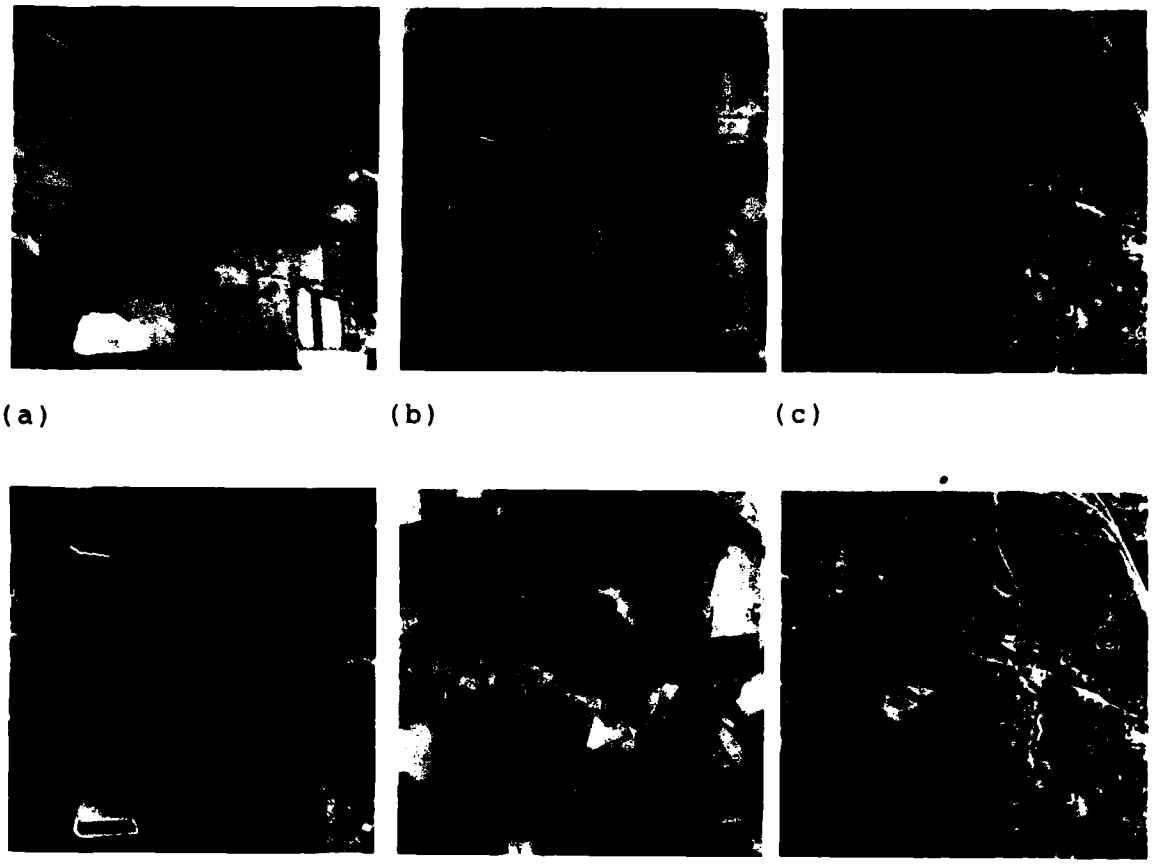
3.2.2 Transformation and Verification of Starting-points

Following the approach mentioned at the beginning of this chapter, the starting-points detected in the 388 x 388 pixels elevation matrix are transformed to the coordinate system of the aerial image by means of the polynomial method (figure 3.12). The chance for a successful centering of the starting-points on the river seems fairly good, because the river is clearly visible in these parts of the image.

Fig. 3.12: Starting-points, displayed in fig. 3.11c, transformed to the aerial image system

The centering of the starting-points can be accomplished by a modified version of the centering of dead-ends developed during the last contract.

In order to perform experiments, it was necessary to generate a file of parameters for easy adaptation. We also assured that the AoIs were enlarged accounting for the fact that an AoI must contain a starting-point as well as a piece of the river for the centering process to be successful. Some results of the experiments performed in the segments of the image matrix reduced by a factor of 4 are displayed in figure 3.13. Due to the local "view" of the extraction methods misinterpretations are unavoidable. Either we accept results on the river as well as on some hedges and forest edges in the neighborhood of the river (figures 3.13a-c) or by selecting a more resticted set of parameter values we get nearly no acceptable result (figures 3.13d-f).



(a) (b) (c)

(d) (e) (f)

Fig. 3.13: Results of tests of centering the starting-points
(a)-(c): Relaxed parameters
(d)-(f): Restricted parameters

Because of the poor centering result, a subsequent extraction does not seem very promising. Therefore we decided to adopt the alternative proceeding: extraction of the valleys from the elevation data by means of the starting-points detected there.

3.2.3 Valley Extraction from the Elevation Data

Because of problems with variable object width, the existing line follower program SUCOUT could only be applied to the reduced elevation matrix of 194 x 194 pixels. To avoid transformation from the 388 x 388 pixels to the 194 x 194 pixels elevation matrix,

the process is setting off from the starting-points detected in the reduced elevation matrix (figure 3.11f).

Besides some minor modifications it was necessary to replace the horizontal evaluation of the REM (regional extraction method). The basic idea of the modification refers to the fact that rivers and creeks are flowing along the bottom of a valley. Therefore, a search for the pixel with the minimum graylevel in each sample line of the AoI (area of interest) seems reasonable. If it is part of a cross-section satisfying condition (3 - 1), then it may be part of a line object, i.e. part of a valley in the present context.

In connection with the reduced elevation matrix we have tested the modified line extraction procedure with different sets of parameter values. We have obtained e.g. the results displayed in figures 3.14 and 3.15a. In these examples of valley extraction the width of the line objects had to range between 1 and 5 pixels and the graylevel contrast had to range between 10 and 20 graylevels. With respect to the result shown in figure 3.15a the width of the line objects ranged between 1 and 4 pixels and the graylevel contrast had the value 10. The CPU time required for extracting the valley came up to 2 minutes. For further processing the result shown in figure 3.15a was selected. To see more details, a magnification to the Comtal format of 512 x 512 pixels is shown in figure 3.15b; figure 3.15c is another representation of the result of the valley extraction. The result of the transformation of the extracted valleys into the coordinate system of the aerial image (figure 3.16a) by means of the polynomial method is used as prediction for water courses by the verification process. Instead of searching over the whole image, only the masked areas displayed in figure 3.16b are left for the verification of water courses.

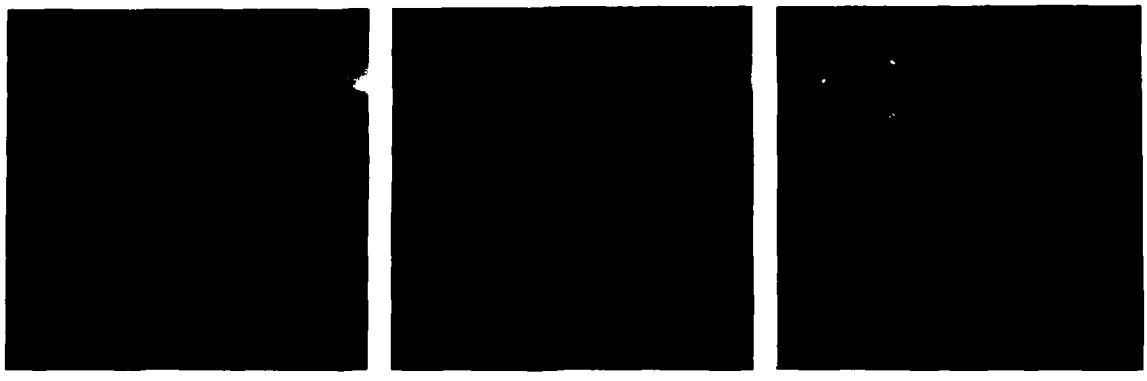


Fig. 3.14: Different valley extraction results from the reduced elevation matrix

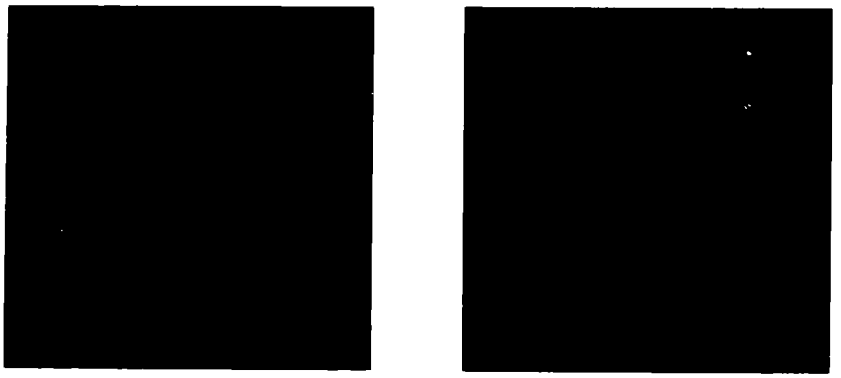
(a) (b) (c)

Fig. 3.15: Different representations of the valley extraction result

(a) 194 x 194 pixels elevation matrix

(b) Comtal format

(c) Area shaped representation



(a) (b)

Fig. 3.16: Prediction of water courses
(a) Transformed result, displayed in figure of 3.15
(b) Areas left for searching for the water courses

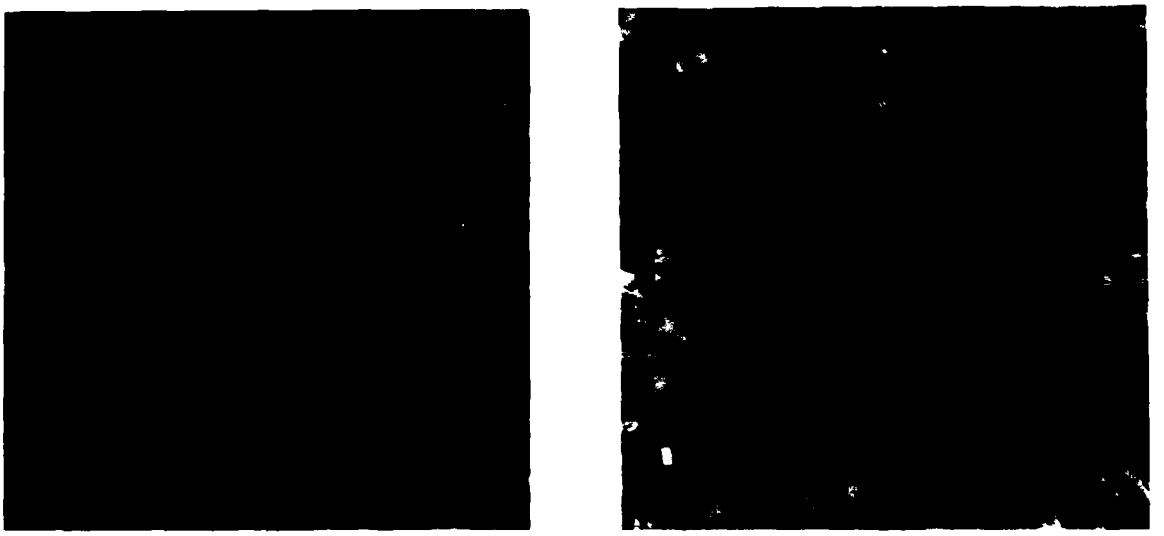
3.2.4 River Verification in Predicted Areas of the Aerial Image

The verification procedure is used to test the prediction that a line object with certain predefined features exists at a given location in the image.

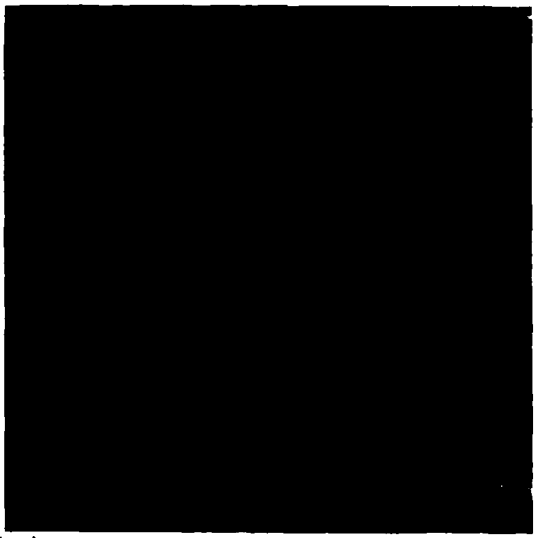
To verify the prediction of water courses, some minor modifications of the stereo verification were necessary /2/. To obtain the width of the object expected by the verification procedure and the REM, the image matrix was reduced by a factor of 4 to the size of 1024 x 1024 pixels. After the modifications and with some parameter adaptations the process of verification has been successfully performed. According to the adjustment of the parameters the modified verification process produced acceptable as well as less acceptable results shown in figure 3.17. Thus for example, with restricted parameters we have obtained the results displayed in figures 3.17a and b and with relaxed parameters the result shown in figure 3.17c. Besides the standard set of parameter values the object width was set to 1 to 4 pixels, and the graylevel contrast was set to the value of 15 graylevels. It took approximately 4 minutes CPU time to verify the extraction result.

To visualize the difference between the modified verification and the original stereo verfication, we have performed a test with each procedure and an identical set of parameter values. Figure 3.18a shows the result when using the original stereo verification and figure 3.18b shows the result of the modified verification process. A comparison of the two results shows greater differences only aside of the main stream. The verification results are similiar. On the basis of the result shown in figure 3.18b it seems not profitable to start a continuation extraction.

To demonstrate the possibility of adding to the verification results, we have performed a continuation extraction setting off from the results shown in figures 18a and b.

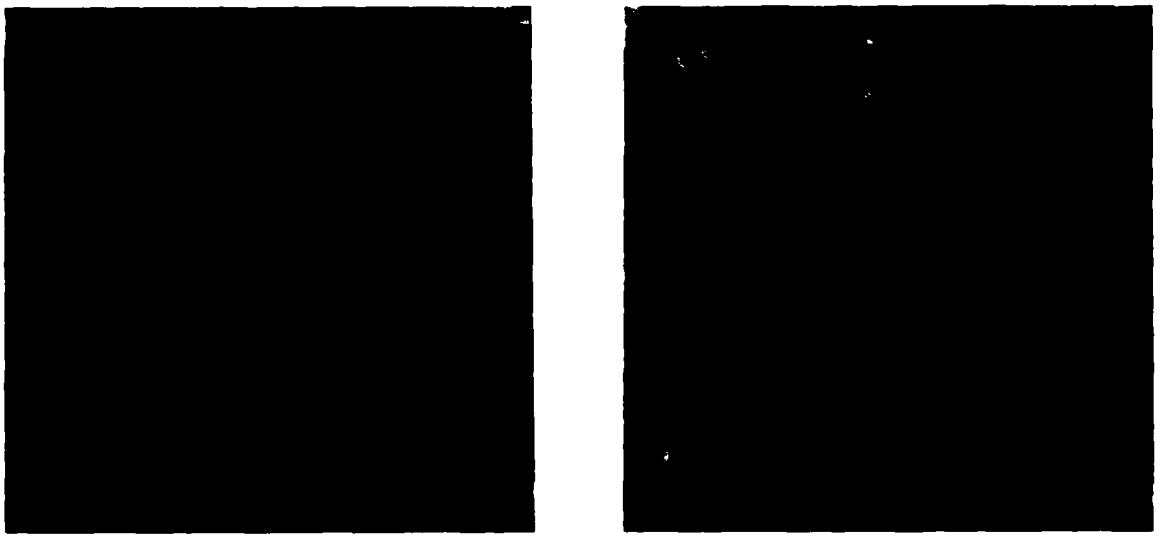


(a) (b)



(c)

Fig. 3.17: Verification results obtained with different sets of parameter values



(a) (b)

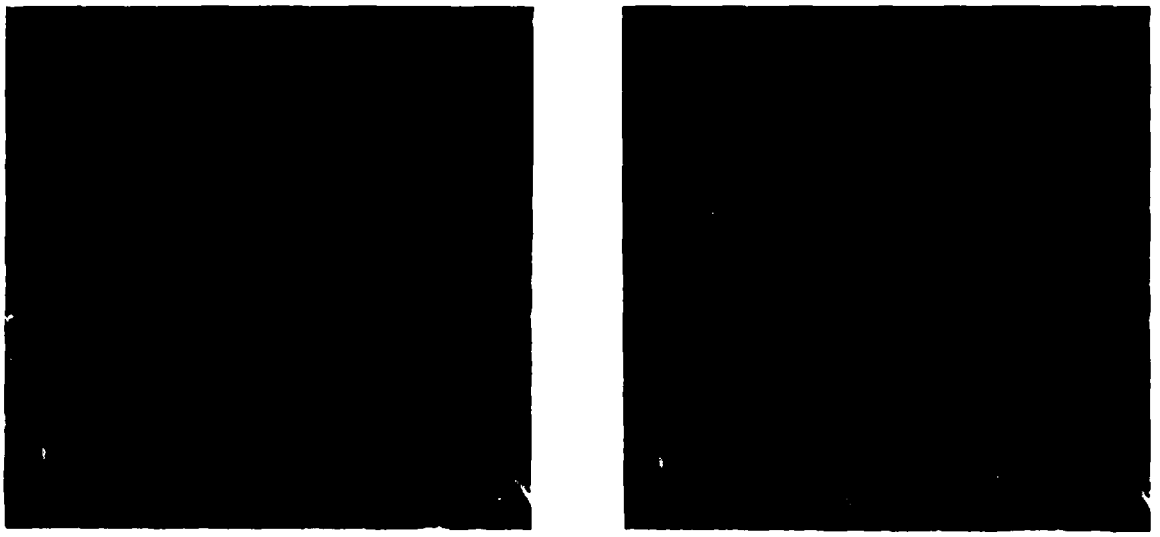
Fig. 3.18: Comparison of the old and the modified verification procedure
(a) Verification result before modification
(b) Verification result after modification

3.2.5 Continuation Extraction from the Aerial Image

The verified course of the river in the aerial image (figures 3.17a and b) shows here and there some gaps. To close these gaps at least partly, the line search program SUCOUT is used given the dead-ends and the result matrix generated by the previous verification process.

Outside the main valley and outside the course of the river some short line objects have been extracted (figures 3.17 and 3.18). Though these short line objects do not form part of a river, they have been detected by the verification process. Most of the detected line objects outside the course of the river are hedges or edges of forests. To concentrate the continuation extraction on the main river we performed a simple threshold operation to delete the short polygons. Figures 3.19a and b show the purged results (compare with figures 18a and b).

Some experiments have been performed with the intention to add to the extraction result using the REM (regional extraction method). The extraction results are displayed in figures 3.20 and 3.21 with the extraction process setting off from the results displayed in figure 3.19a and b, respectively. The set of parameter values was the same for both result series. The values 10 to 25 were assigned to the graylevel contrast and the required object width was set to 1 to 5 pixels. The results displayed in figures 3.20a and b show that the first continuation extraction attempt did not succeed in detecting the correct course of the river in the urban area. The second attempt, displayed in figure 3.21, was more successful. The sharp bended loop, approximately in the middle of the course of the river, could not be handled by the existing methods.



(a) (b)

Fig. 3.19: Purged verification results
(a) Purged result (compare with figure 3.17a)
(b) Purged result (compare with figure 3.17b)

(a) (b)

Fig. 3.20: Continued extraction results (compare with figure 3.19a)

(a) (b)

Fig. 3.21: Continued extraction results (compare with figure 3.19b)

3.3 Discussion

Difficulties to detect the exact course of the river automatically must be admitted for the urban area, where even a human observer of the aerial image cannot be sure to see the river. The same thing happens with respect to the continuation of the upper end of the extracted course of the river, where the valley is very narrow. It is also not possible to verify or extract the sharp bended loop of the river with the existing methods.

The cooperative extraction methods have proved altogether successful in extracting the course of a river, though difficulties arise from the fact that the extraction methods refer to man made line objects like roads. Thus, general predictions about the course of a river are less reliable than about man made objects. For example, the flow direction of a river may change within a short distance, the width of a river may be changing significantly and the appearance of forest edges or of hedges may be very similar to the appearance of a river.

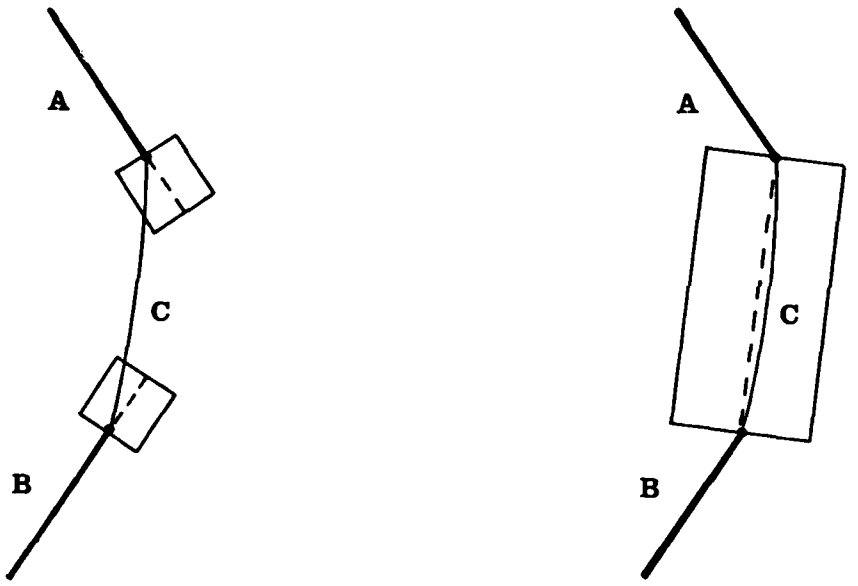
4. Improvement of Road Extraction

4.1 General Idea

Methods for the automatic extraction of line objects from aerial images have been developed in previous projects /1,2/. The performance of the methods has been demonstrated by the extraction of line objects representing road networks. The incompleteness of the extraction results is partly due to the "conservative" behaviour of the extraction procedures which demand a rather continuous appearance of the line objects. Extensive distortions result in the termination of the extraction process. The extraction procedures follow the line objects sequentially. They inspect their local neighborhood, but do not account for the global context. Therefore, they do not provide enough infomation for reliably extracting line objects of poor appearance. In consequence, line segments may be missing in the extraction result. Examples of two initial extraction results and two coincident extraction results of image No. 56 and image No. 54 are displayed in figures 4.20a, 4.21a, 4.22a and 4.23. The extracted line networks are superimposed on the graylevel images. The gaps are clearly visible. At some of these places the failure of the extraction process is intelligible, since there is no line continuation visible. At some other places it seems worth while thinking about a new approach to extract the missing line segments, since the respective object segments are visually detectable.

We have developed an improved extraction strategy from the following idea: If we were able to determine those gaps in the extraction result, where two or more line segments are pointing at each other, we could define AoIs (Areas of Interest) between the ends of the respective object segments and try to extract the missing line segments by the REM (Regional Extraction Method). The position and orientation of the AoI would be more promising than they were, if we had approached the gap from one side only. From figures 4.1a and b the difference between the old and the new approach, a and b respectively, is evident. The chance to detect a piece of a line object, not necessarily the complete miss-

ing segment C, between the two already extracted segments A and B is obviously better in the case of figure 4.1b than in the case of figure 4.1a. This is particularly evident, if there are distortions of the appearance of the object near the ends of the segments A and B. The AoI in figure 4.1b covers the relevant area and adjusts to the orientation of the missing segment much better than the two AoIs in figure 4.1a. The orientation of the AoI in figure 4.1b is practically identical with the general orientation of the line segment C. The situation in figure 4.1a is not as favourable.



(a) (b)

Fig. 4.1: Old (a) and new (b) approach

It was the aim of this part of the project to complete the extraction results at such places by integrating more a priori knowledge about road networks into the extraction process. We have addressed the idea of using a priori knowledge about geometric features of road networks in a first approach and of adding some a priori knowledge about the ways road networks are embedded into the topography in a second approach to extract object parts still missing.

Since as a rule a road network does not tend to have too many gaps, the existence of any two neighboring, opposite line segments of the extraction result pointing at each other suggests a missing line segment between them. Thus, in view of the geometry of road networks, distance and collinearity of line segments may serve as criteria for predicting a continuation extraction at gaps of the extracted line network. Regarding the embedding features of road networks into the topography, a segmentation of the image data covered by the AoI and a classification of the regions in it would support even more conclusions. E.g., a straight line connection between the extracted object parts by an object edge, like the edge of a forest, could serve as a guide for the extraction of the missing line segments. The abscence of such an edge on the other hand and the presence of an obstacle, like a lake, between both parts would induce a definite termination of the extraction process at that place. Figure 4.2 shows a situation where geometry-driven considerations may lead to the prediction of a missing line segment between the extracted segments A and B due to their orientation and the length of the interval between them. Additionaly, image-context- driven considerations may support this prediction, since object edges connect the ends of the segments: the edge of a wood, labeled by "W", and the edges of agricultural areas, labeled by "Ai".

We have integrated these ideas into an improved strategy for the extraction of line objects:

(1) initial extraction of the road network,
(2) determination of the CALCs (candidates for line continuation),
(3) grouping of the CALCs,
(4) verification, (extraction of predicted line segments between grouped CALCs),
(5) region analysis,
(6) contour analysis,
(7) verification.

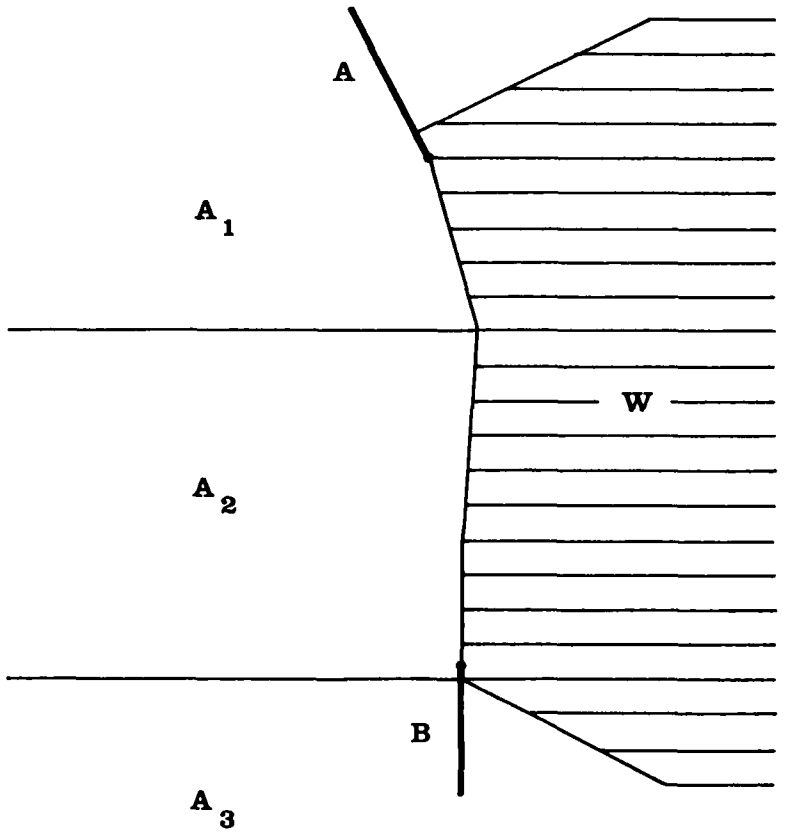


Fig. 4.2: Geometry and image-context aspects

The sequence of 1 to 4 has been successfully developed, tested and implemented. The sequence of 5 to 7 has been studied. Both sequences will be discussed in detail in the sequel.

4.2 The Geometry-Driven Approach

4.2.1 The CALCs

There are different types of candidates for a continuation extraction. Since a candidate is a line segment of the kind involved in the constitution of a gap in the extracted line network, they can be inferred from the definition of such a gap.

Definition: A gap in the extracted line network is the empty space between two line segments A and B such that A is pointing at B or B is pointing at A.

From the definition we may infer four types of candidates:

(1) dead-ends, i. e. isolated ends in the extracted line network,
(2) sharp bends, i. e. parts of the extracted line network with a significant amount of curvature,
(3) crossings, i. e. locations in the extracted line network where several line segment meet,
(4) projection-points, i. e. those segments in the extracted line network at which dead-ends, branches of sharp bends or branches of crossings are pointing.

Strictly speaking, it is not the bend or the crossing that counts as a candidate but its branches. A typical situation is sketched in figure 4.3 with the CALCs labeled by the first letter of their names. It may also happen, as in the case of the bend B1 and the projection-point P1 in the sketch, that two or more CALCs claim the same segment.

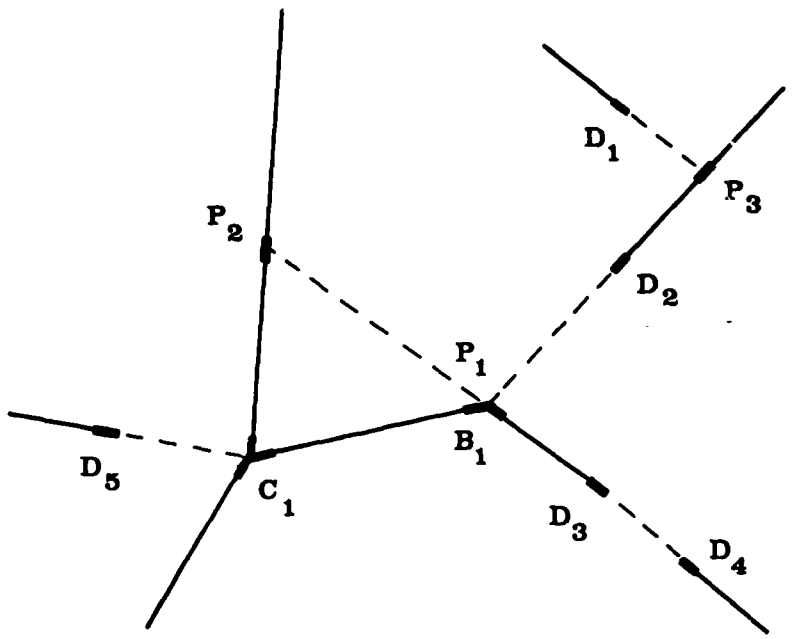


Fig. 4.3: A typical situation with CALCs

Given the four types of candidates, we can conceive of nine different combinations of two candidates each satisfying the definition of a gap, figures 4.5a-i:

(a) two dead-ends,
(b) a dead-end and a branch of a bend,
(c) a dead-end and a branch of a crossing,
(d) a dead-end and a projection-point,
(e) two branches of two different bends,
(f) a branch of a bend and a branch of a crossing,
(g) a branch of a bend and a projection-point,
(h) two branches of two different crossings,
(i) a branch of a crossing and a projection-point.

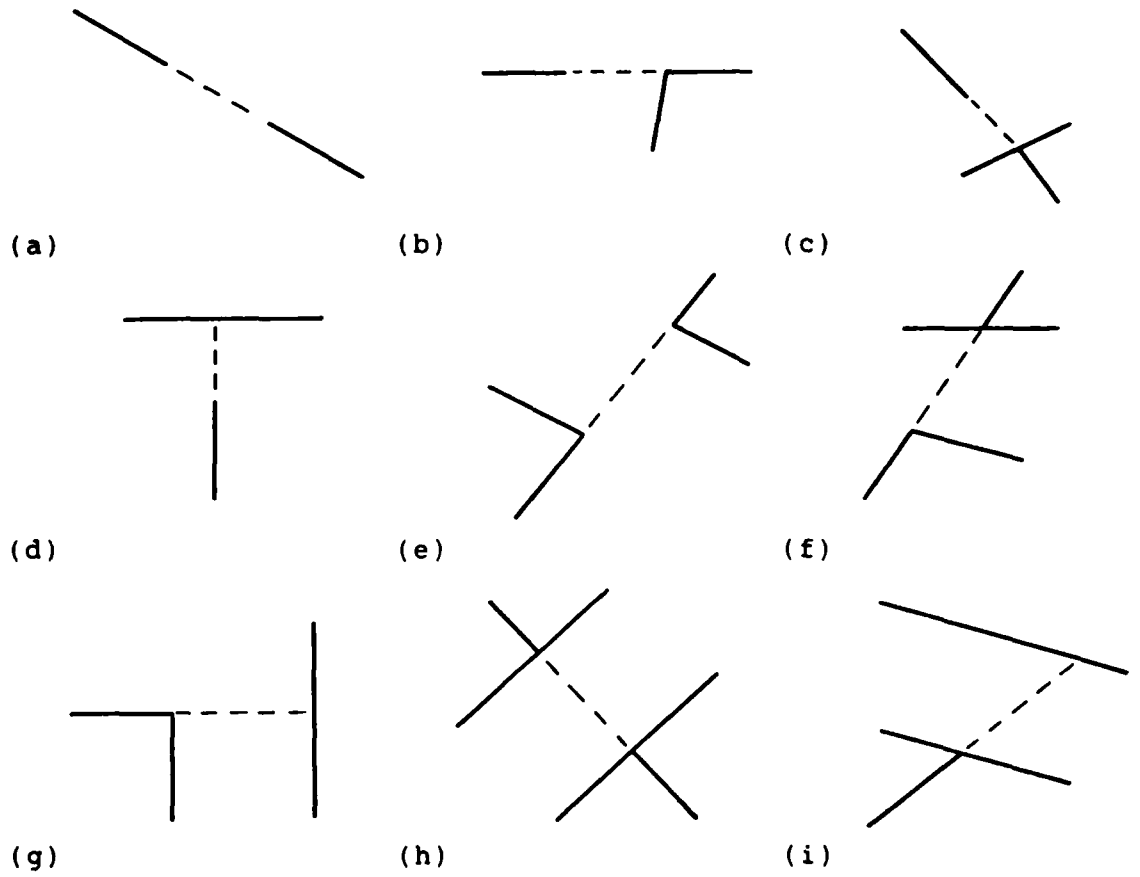


Fig. 4.4: Types of gaps

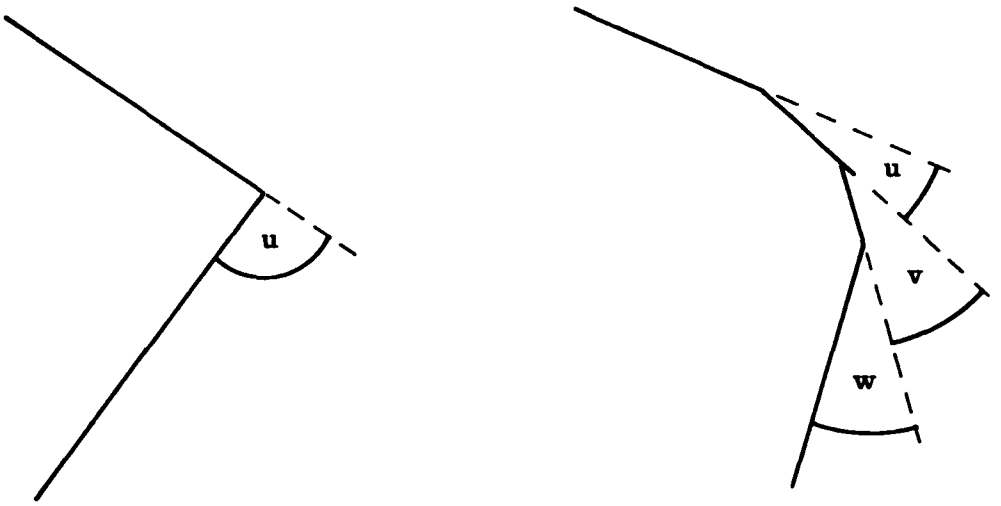
After the definition of the different types of CALCs, we can turn to the problem of determining their locations in the extracted line network.

Each extraction of line objects from the graylevel image is followed by a result matrix extraction of maximum length polygon lines. If the end segments of a polygon line are isolated, they are called "dead-ends". If two or more of them have the same end vertex, they are called "branches of a crossing". In any case they are easily accessible by scanning the maximum length polygons and comparing the end vertices.

The detection of sharp bends and their branches is more complicated. In the first place, we have to distinguish between two types of sharp bends:

(1) If the angular difference of orientation between two adjacent segments is greater than a threshold, we are facing a sharp bend of the first kind. This criterion is adequate for sharp bends formed by two segments enclosing an acute angle. It is not adequate for sharp bends consisting of more than two short segments enclosing obtuse angles.
(2) If the average curvature of two or more adjacent segments is greater than a threshold, we are facing a sharp bend of the second kind. "Average curvature" denotes the ratio of the sum of the angular differences of orientation of adjacent segments to the sum of their lengths. This criterion is adequate for sharp bends formed by more than two short segments enclosing obtuse angles, but it is not adequate for sharp bends consisting of two long segments enclosing an acute angle.

A comparison of both types of bends in figures 4.5a and b illustrates the different concepts. The bend sketched in figure 4.5a requires only the calculation of the angle u and the comparison of its absolute value with a threshold. This procedure cannot be adequate for bends like the one sketched in figure 4.5b. There, each single angle u,v and w is neglectable, but their sum is substantial. Thus, though each angle on its own does not indicate the existence of a sharp bend, the sum of them may do so. However, the existence of a sharp bend does not only depend on the sum of the angles, it depends as well on the lengths of the line segments involved. If the segments are long, we are less inclined to think of a sharp bend than when they are short. By analogy with the concept of curvature in continuous geometry we define in the discrete case the average curvature of a bend to be the ratio of the sum of the angles involved to the sum of the lengths of the segments involved. Obviously, this concept applies to the bend in figure 4.5b but not to the one in figure 4.5a. Hence, we need both concepts for the detection of all bends.



(a) (b)

Fig. 4.5: Two types of bends

The detection of sharp bends of the first kind is as easy as the detection of dead-ends or branches of crossings. The polygon lines are scanned and the angular difference of orientation between every two adjacent segments is computed and compared to a threshold. If it is greater than that threshold, then the two segments constitute a sharp bend and are its branches. The detection of sharp bends of the second kind is not as easy. The polygon lines must be scanned from one end to the other. For each pair of adjacent segments the direction of traversal has to be computed. It may be clockwise, counter-clockwise or straight. The location of reversal of the direction of traversal indicates the end of a bend. For each substring of segments within the bend the average curvature is determined. If it is greater than a threshold, the segments involved are labeled. If there are any labeled segments after the bend's complete processing, the branches in question are the unlabeled direct neighbor segments of the outermost labeled segments of the bend. For an illustration of the extraction of bends from a polygon see figures 4.6 and 4.7. Scan-

ning the polygon from A to D, the vertices of the polygon are labeled "0", "-", "+" and "*", denoting straight, clockwise and counter-clockwise direction of traversal and begin or end of the polygon, respectively, figure 4.6. In order to distinguish between straight, clockwise and counter- clockwise direction of traversal, the cross product of every pair of adjacent segments is calculated, with the segments viewed as vectors. A positive cross product corresponds to counter-clockwise, a negative to clockwise and the zero vector corresponds to straight direction of traversal. To find the boundaries of a bend in the polygon, a finite automaton is used, see figure 4.7 for its state transition diagram. The automaton has two final or accepting states q0 and q3. However, only the arrival at q3 proves, that a bend has been traversed. As an example for the way the automaton works take the polygon drawn in figure 4.6. Starting with the input of "*" at A, the process stops with the input of "+" at C, the right boundary of the first bend of the polygon. Reversing the direction of traversal at C, starting with the input of "0" to the left of C, the process stops with the input of "*" at A, the left boundary of the bend. Again, starting with the input of "+" to the right of C, the process stops with the input of "*" at D, the right boundary of the second bend of the polygon. Reversing the direction of traversal at D, starting with the input of "0" to the left of D, the process stops with the input of "-" at B, the left boundary of the bend. In this way both bends are extracted from the polygon.

The detection of dead-ends, branches of sharp bends and crossings must precede the detection of projection-points, since they are needed for the latter's detection. See figure 4.8 for the following explanation of the procedure. Given a candidate C, a window W is defined. Several half lines are computed starting at the end vertex of the candidate. They are bound to the window, to a given angular distance u between each other and to a given maximum deviation of orientation MA from the orientation of the candidate. If they intersect with opposite polygon segments within the window, a confidence value c is computed for each point of intersection. It depends on the point's distance d from the end of the

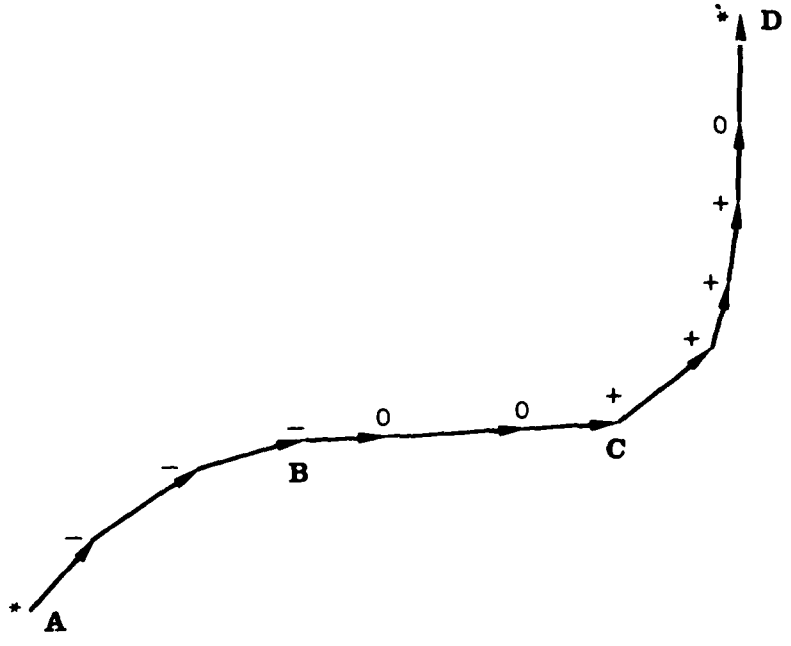


Fig. 4.6: Labeled bend

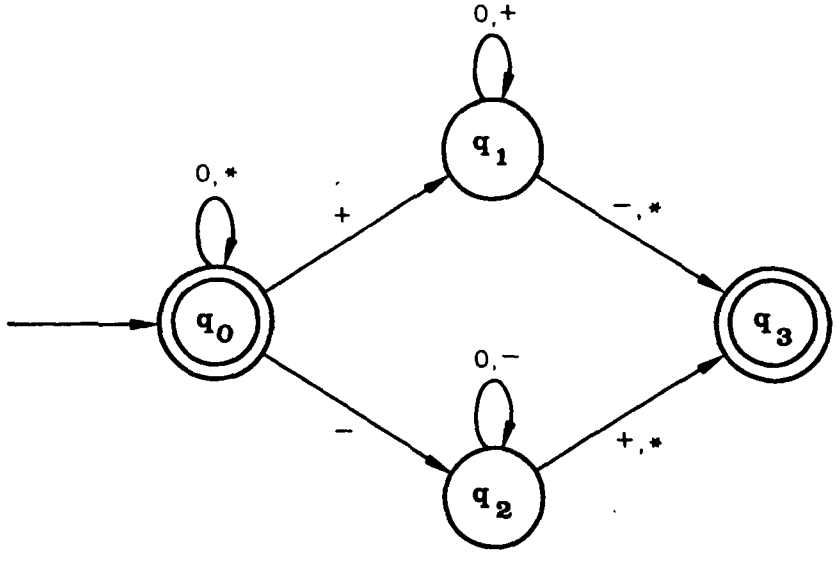


Fig. 4.7: Bend detection automaton

candidate and on the deviation v of the orientation of the intersecting line from the orientation of the candidate, given the maximum deviation permissible MA and the maximum distance permissible MD. The confidence is a function of the distance and the

deviation of orientation, decreasing with increasing distance and increasing deviation of orientation.

$$c = \begin{cases} 1-(|v|/|MA| + d/MD)/2, & \text{if } MA \neq 0 \\ 1-d/MD, & \text{if } MA = 0 \end{cases} \qquad (4 - 1)$$

Formula 4 - 1 is just one way of computing a confidence value for a point of intersection. It is not the only possible way, but the one we have chosen. The ratio of the actual deviation of orientation to the maximum deviation of orientation, $|v|/|MA|$, and the ratio of the actual distance to the maximum distance, d/MD, are weighted equally. The intersection point with the highest confidence value is called "projection-point" of the actual candidate.

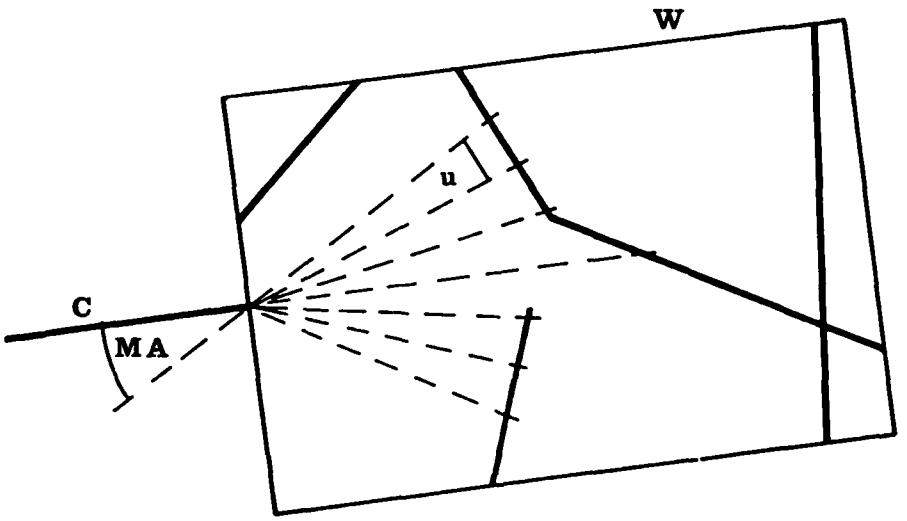
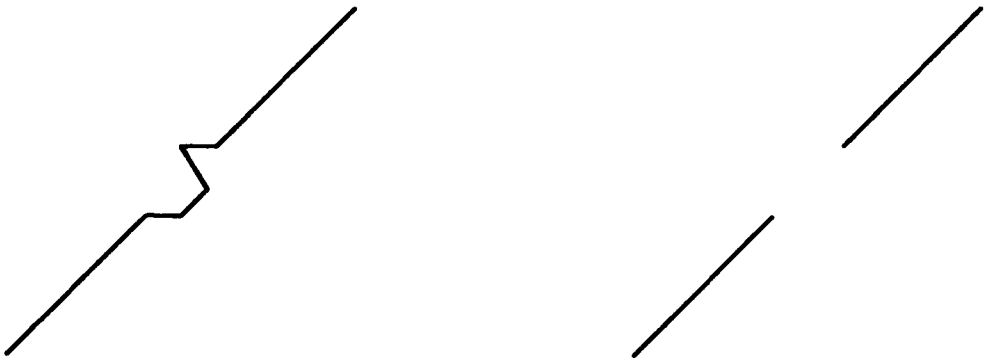


Fig. 4.8: Calculation of projection-points

Before proceeding to the discussion of the grouping procedures, a final remark about the use of the procedures for the detection of bends may be right in time. Due to the local "view" of the extraction methods, there may be some irregularities in the extracted line network. The procedures for the detection of bends can contribute to the correction of these deviations by detection

and elimination of the respective line segments, figures 4.9a and b. By deleting such irregularities gaps and dead-ends are generated, which may be included into the whole process of determining the CALCs, grouping the CALCs and verifying the predicted line segments between the CALCs. Thus, the irregularities may be corrected in the next verification process.



(a) (b)

Fig. 4.9: Polygon before (a) and after (b) purging

4.2.2 Grouping of the CALCs

The aim of the grouping process is to determine those gaps in the extracted line network, where missing line segments may be expected. This is done by pairing the candidates according to distance and collinearity. A particular grouping process has already been described by explaining the association of projection-points with dead-ends, branches of sharp bends and crossings.

A different grouping process is required for the pairing of candidates from the set containing the dead-ends, the branches of sharp bends and the branches of crossings. Not every possible combination of candidates is acceptable as a prediction for a continuation extraction. A pair of candidates A and B is selected only if the following conditions are satisfied, figure 4.10:

(1) The distance d between the candidates is less than a threshold.
(2) The absolute value of the angular difference of orientation u of the candidates is not greater than the actual, absolute value of a(d).
(3) The absolute value of the angular difference of orientation v or w of the candidates and the straight line connection between them is not greater than the actual, absolute value of a(d).
(4) The confidence value c for a pairing is not less than a threshold.
(5) No line segment already extracted intersects the straight line connection of the candidates.

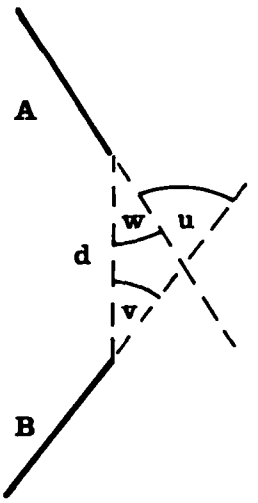


Fig. 4.10: Grouping measures for two CALCs

The first condition establishes a maximum distance acceptable for all candidate pairs. The second and third condition establish an upper limit for angular differences of orientation, the limit being a function |a(d)| of the actual distance d between the candidates.

$$|a(d)| = |Ma - (Ma/Md)d| \qquad (4 - 2)$$

For the graph of |a(d)| in polar coordinates see figure 4.11. Its maximum value equals |Ma|, which is reached at the distance zero when the candidates touch. The absolute minimum of the limit with the value zero is at the maximum distance permissible between any two candidates. The limit is a continuously decreasing function in the interval bounded by the distance zero and the maximum distance permissible. From the graph in figure 4.11 it is clear that with increasing distance the tolerable difference of orientations between two CALCs decreases.

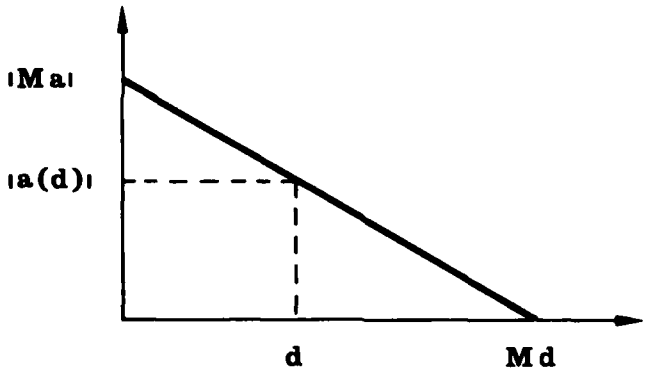


Fig. 4.11: Graph of |a(d)| in polar coordinates

It is instructive to transform the graph of a(d) into cartesian coordinates, figure 4.12. Let a candidate be a line segment on the x-axis and let the origin of the coordinate system be its right vertex, then the line L defines the boundary of the neighborhood region NR of the candidate. Condition 3 above implies, that two candidates are neighboring only if one of them is lying within the other's neighborhood region. Additionaly, condition 2 imposes a threshold on the difference of orientation between the two candidates. Hence, referring to the sketch in figure 4.13, segments B and D are neighbors of A and segments C and E are not. The pair (A,C) does not satisfy condition 3 and the pair (A,E)

does not satisfy condition 2.

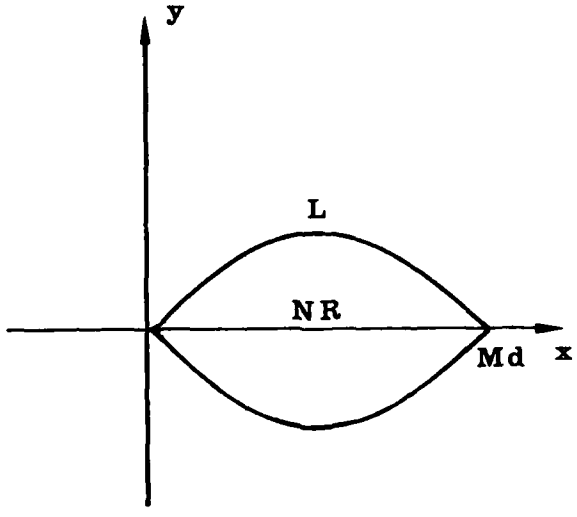


Fig. 4.12: Graph of a(d) in cartesian coordinates

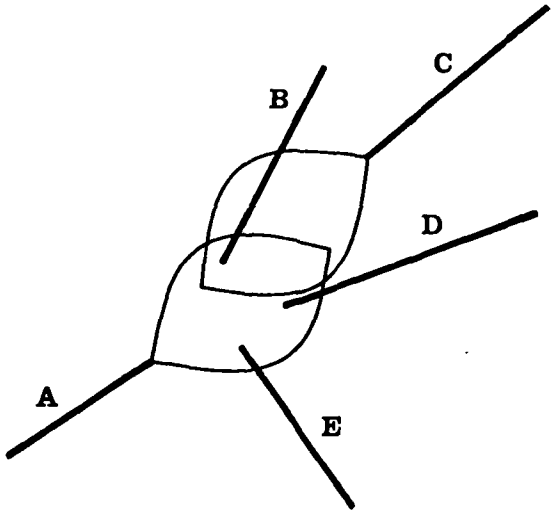


Fig. 4.13: Grouping of CALCs

A candidate may have several neighbors which satisfy the first three conditions. Usually they do not satisfy them likewise well. Hence, a natural ranking between the neighbors of a candidate may be inferred for the subsequent extraction process. The candidate which satisfies the conditions best is considered first for predicting missing line segments. The criterion for ranking the neighbors of a candidate is the confidence c defined as

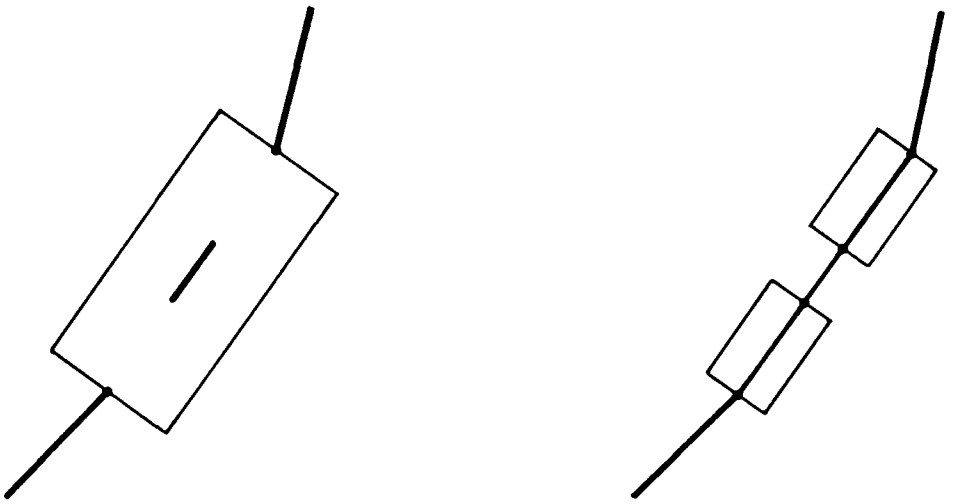
$$c = 0.5((|a(d)|-|u|)+\max(|a(d)|-|v|,|a(d)|-|w|)). \qquad (4-3)$$

What has been said before about formula 4 - 1 applies to formula 4 - 3 too. It is just one way to compute the confidence values in question. They are calculated from the three angular differences $|a(d)|-|u|$, $|a(d)|-|v|$ and $|a(d)|-|w|$.

4.2.3 Verification of Predicted Line Segments

The output of the grouping process is an ordered set of paired candidates between which missing line segments are predicted. To verify these segments in the graylevel image, the REM is put into action. The REM itself has not changed since the last project. However, the conditions of its use have altered. They have been changed, in order to account for the needs of this verification process. The position of the AoI does no longer depend on the orientation of the actual candidate, but instead on the orientation of the line segment connecting the actual candidate with its neighbor; it joins the two candidates, figure 4.1b. Since the prediction of missing line segments has become more reliable due to the grouping process, we have decided to suspend another constraint imposed on the verification process. A new piece of line extracted between two paired CALCs need not be connected to either of them, figure 4.14a. If there are gaps left to either side of the line, it may nevertheless be accepted. It is possible to iterate the whole sequence of determining the CALCs, grouping the CALCs and verifying predicted line segments between the CALCs, in order to close the gap completely. The REM selects only the best line detected in the AoI and this line need not be complete. In

that case, a further attempt to close the gap completely may be successful, since it starts from improved assumptions, figure 4.14b. The gap has become smaller. Hence, the same is true for the AoI defined there. So, the set of concurrent locations for a line continuation in the new AoI may be smaller than the old one.



(a) (b)

Fig. 4.14: Iterative verification,
first (a) and second (b) continuation extraction

4.2.4 Results

To demonstrate the performance of the system, we have chosen two aerial images labeled image No. 54 and 56 from the image data provided by USAETL. The digitized graylevel images were already available since the last project. We have selected a 4000 x 4000 pixel section from each image for processing, figures 4.15 and 4.16. Both sections were subdivided into 8 segments of 1024 x 1024 pixel each. The segments are indicated by the grids of figures 4.15 and 4.16.

It is not possible to visualize all details of the processing se-

quence for a section as a whole in this report, due to lack of space. Therefore, we have chosen three particularly interesting segments from the 4000 x 4000 section of image No. 56 to illustrate the details. Results of the whole sections are presented subsequently.

Before discussing the results, let us recall the main blocks of the processing sequence and describe some parameters involved, always remembering that the parameter values must adapt to the image data and that there might be more than just one set of appropriate parameter values for the respective process. The parameter values chosen have emerged from tests with the data from image No. 54 and 56.

The initial extraction of the road network in the graylevel image is the first block in the processing sequence. We have obtained the initial extraction results shown in the sequel by the standard set of parameter values with the exception of two values which were chosen a little more restrictive:

(1) The width of the object, which had to range between 2 and 4 pixels.
(2) The grayvalue contrast of a cross section of the object, which was required to be at least 20 grayvalues.

The determination of the CALCs is the second block in the processing sequence. It comprises the detection of dead-ends, branches of sharp bends and branches of crossings and the generation of projection-points.

For the detection of sharp bends there are two important parameters corresponding to the two types of sharp bends described above, see section 4.2.1:

(1) The first parameter refers to the minimum angular difference of orientation between two adjacent line segments as a criterion for being a sharp bend. It was set to 45 angular degrees.

(2) The second parameter refers to the minimum average curvature of a bend as a criterion for being a sharp bend. It was set to 2.15 angular degrees per pixel. Naturally, the value of this parameter depends on the degree of approximation or smoothing of the extracted lines. The value of 2.15 has proved effective if we allow a distance of up to 3 pixels between the extracted line and the approximating line segments.

Regarding the generation of projection-points, there are several restrictive parameters:

(1) The size of the window where the search for crossing line segments takes place. It was set to 300 x 40 pixels. The reason for using a window is a pragmatic one. We could have done completely without a window, but we wanted to reduce the amount of calculations.
(2) The angular difference of orientation between adjacent half lines from the end of the CALC. It was set to 1 angular degree. We have chosen a high sampling frequency, in order not to miss any important line segment.
(3) The maximum angular difference of orientation between the CALC and the half lines. It was set to 20 angular degrees. This parameter fixes the bounds for sampling. Its value should be kept small, so that it will not come into conflict with the concept of projection.
(4) The maximum angular difference of orientation between crossing line segments and the normal of the CALC. It was set to 55 angular degrees. The introduction of this parameter was meant to suppress the calculation of projection-points on collinear line segments.

The grouping of the CALCs is the third processing block. Here, the important parameters are those which specify the neighborhood function $|a(d)|$:

(1) The maximum distance between two CALCs. It was set to 500 pixels. If the CALCs are crowded, the value of this parameter may be kept low without loss. If not, there is nothing to be

said against very high values of the parameter.

(2) The maximum angular difference of orientation between two CALCs and between a CALC and the straight line connection of the CALCs, respectively. It was set to 45 angular degrees. This parameter is a limit for the degree of collinearity required. Therefore, its value should not be too high.

The verification of the predicted line segments between grouped CALCs is the last block within the processing sequence of the geometry-driven approach. It amounts to an extraction by the REM. Hence, its set of parameters is a subset of the set of parameters of the initial extraction process. Two parameter values chosen should be mentioned:

(1) The width of the object, which had to range between 2 and 6 pixels.
(2) The grayvalue contrast of a cross section of the object, which was required to be at least 12 grayvalues.

Compared to the respective parameters in the initial extraction process, the two parameters used in the verification process have been chosen less restrictive. The aim was to extract not yet detected road segments, perhaps of poor appearance, on the basis of improved predictions with relaxed parameters.

Next, we present single 1024 x 1024 pixel segments from the 4000 x 4000 pixel section of image No. 56 and results from processing them.

Segment No. 2:

Figures 4.17a-f illustrate a complete processing sequence. Figure 4.17a shows the graylevel segment before processing. A close look at the picture reveals agricultural areas, forests, a river and part of a road network with three crossings, of which just the middle one is clearly visible. Figure 4.17b shows the graphic result of the initial extraction process superimposed on the graylevel segment. Only part of the road network has been extracted.

The determination of the CALCs, including the elimination of irregularities from the initial extraction result, leads to the line network shown in figure 4.17c. The extracted crossing and the sharp bends have been transformed to a set of new, preliminary dead-ends, in order to accomodate to the grouping process. The result of the grouping process is shown in figure 4.17d with the grouping indicated by white, dashed lines. The result of the verification process is shown in figure 4.17e. Those short line segments, which merely represent a straightening of the initial extraction result, are not very important. However, the long piece of line in the upper, right part of the segment clearly demonstrates the efficiency of the geometry-driven approach. The complete, final result is shown in figure 4.17f. It was not possible to extract the upper crossing completely. The upper left branch was not detected. The appearance of the road there is so bad, that it seems completely impossible to have any success without using some semantics and interpretation, things that lie beyond the scope of this project.

We have mentioned earlier, that the whole sequence of determining the CALCs, grouping the CALCs and verifying the predicted line segments between the CALCs can be iterated as long as anything significant is added to the extraction result. In the present case the iteration of the process does not add significantly to the result. An example for an iterative completion of the extraction result is shown in the next sequence.

Segment No. 3:

Figures 4.18a-f show, like figures 4.17a-f before, the results of the extraction process, beginning with the graylevel segment, figure 4.18a, and ending up with the initial extraction result extended by the first verification result, figure 4.18f. Again, the initial extraction process does not succeed in extracting all road segments visible, figure 4.18b. It leaves us with a wide gap in the lower middle part of the segment. Due to the poor appearance of the road at that location, the line forming the upper boundary of the gap departs from the correct course of the road

and points into the wrong direction. Only the calculation of the CALCs, including the elimination of irregularities, figure 4.18c, and the grouping of the CALCs, figure 4.18d, establish the correct relations. The result of the subsequent verification process is shown in figure 4.18e. An important piece of line has been extracted in the lower part of the segment. However, the big gap in the respective part of the extraction result has not been completely bridged. Hence, an iteration of the whole process is initiated, starting from the actual set of extracted polygons, determining the CALCs in this set, grouping them and verifying the predicted line segments. The whole sequence was iterated four times until the gap was completely closed. Figures 4.18g-j show the gradual completion of the extraction result after each single iteration.

Segment No. 8:

The last example represents a rather complex situation, figures 4.19a-i. Figure 4.19a shows the graylevel segment before processing. The initial extraction result is shown in figure 4.19b. It contains a lot of gaps and some significant deviations from the correct course of the roads. The elimination of some irregularities and the calculation of the CALCs result in a more satisfactory line network, figure 4.19c. In the foregoing two examples we did not illustrate the generation of the projection-points, since they did not contribute to the process significantly. But in the present case their influence on the verification result is even stronger than that of the other CALCs. Hence, we illustrate their generation, figure 4.19d, and the grouping of the other CALCs, figure 4.19e, seperately. Consequently, the result of the verification process referring to the projection-points, figure 4.19f, and the result referring to the other grouped CALCs, figure 4.19g, are displayed seperately too. Together with the purged initial extraction result they are shown in figure 4.19h. The final result after two iterations of the whole process is shown in figure 4.19i. There are still some gaps left in the extracted line network, but the progress is evident.

We have also processed the section as a whole. The results of the initial extraction and the continued extraction are displayed in figures 4.20a and b, respectively. A look at both figures reveals that the geometry-driven extraction has added significantly to the result of the initial extraction. This is particularly evident for the sample segments discussed above, i.e. for segments No. 2, 3 and 8. A completion of the extraction result can be also observed at some other locations, e.g. segments No. 10 and 11.

We have also processed the 4000 x 4000 pixel section from image No. 54. The results of the initial extraction and the continued extraction are displayed in figures 4.21a and b. The improved procedures were particularly successful in segments No. 1, 4, 9, 10 and 11.

As indicated by the processing of the sample segments, the results of both sections can be still improved by iterating the process as long as anything significant is added to the extracted line network.

To get an impression of the time requirements for the sequence of determining the CALCs, grouping the CALCs and verifying the predicted line segments between the CALCs, we have measured the processing times for the section from image No. 56 using the set of parameter values described above. The approximate time requirements were as follows:

(1) 3.5 CPU minutes were consumed by the determination of the CALCs. Most of the time was needed for the calculation of the projection-points, about 3 minutes, due to the high sampling frequency in scanning the neighborhood of each CALC.
(2) 1 CPU minute was required for grouping the CALCs.
(3) 35 CPU minutes were consumed by the verification of the predicted line segments between the CALCs. The elapsed time, i.e. the time consumed from the start of the verification job until the end of the job came up to 40 minutes. The small difference between elapsed and CPU time is due to the use of the virtual addressing mode in accessing the elements of the

image matrix and the result matrix (a binary matrix containing the extracted lines). Reading matrix sections from file and writing matrix section to file is accomplished by system services and need not be taken care of by the program.

4.2.5 Integration of the Improved Extraction into the Procedure of Coincident Line Extraction

The new system may be used whenever a line network has been extracted from a graylevel image and is given in the form of a set of polygons. The procedure of coincident line extraction from stereo image pairs, which has been developed during the last project, produces such results after each main step of the process, i.e. after each extraction of line objects and after each verification of the extracted line objects. For a detailed discussion of the procedure see /2/. The main steps of a coincident line extraction from a stereo image pair A and B, starting e.g. with image A, are:

(1) the (initial) extraction of line objects from A,
(2) the verification of the extracted line objects from A in B,
(3) the (continuation) extraction of line objects from B,
(4) the verification of the extracted line objects from B in A,
(5) the (continuation) extraction of line objects from A.

The sequence of 2 to 5 can be iterated until no further completion of the extraction result is possible.

The new extraction procedure may be integrated into the process after each extraction or verification. Two examples:

The initial extraction and the geometry-driven extraction from image No. 56, discussed above, could be regarded as the first step of a coincident line extraction process to be continued by a transformation and verification of the extraction results in image No. 54.

After the termination of the last project, we nave performed another coincident extraction from images No. 54 and 56 with slightly modified parameter values. The results are displayed in figures 4.22a, for image No. 56, and 4.23, for image No. 54. An attempt to extract line segments still missing from image No. 56 by the new procedure, resulted in the line network of figure 4.22b. Some gain can be observed in segments No. 8, 9 and 11. Of course, in order to conform to the concept of coincident line extraction, the result cannot be accepted before having been verified in image No. 54. However, without presenting the verification result it should be evident, that the new method can be integrated into the concept of coincident line extraction.

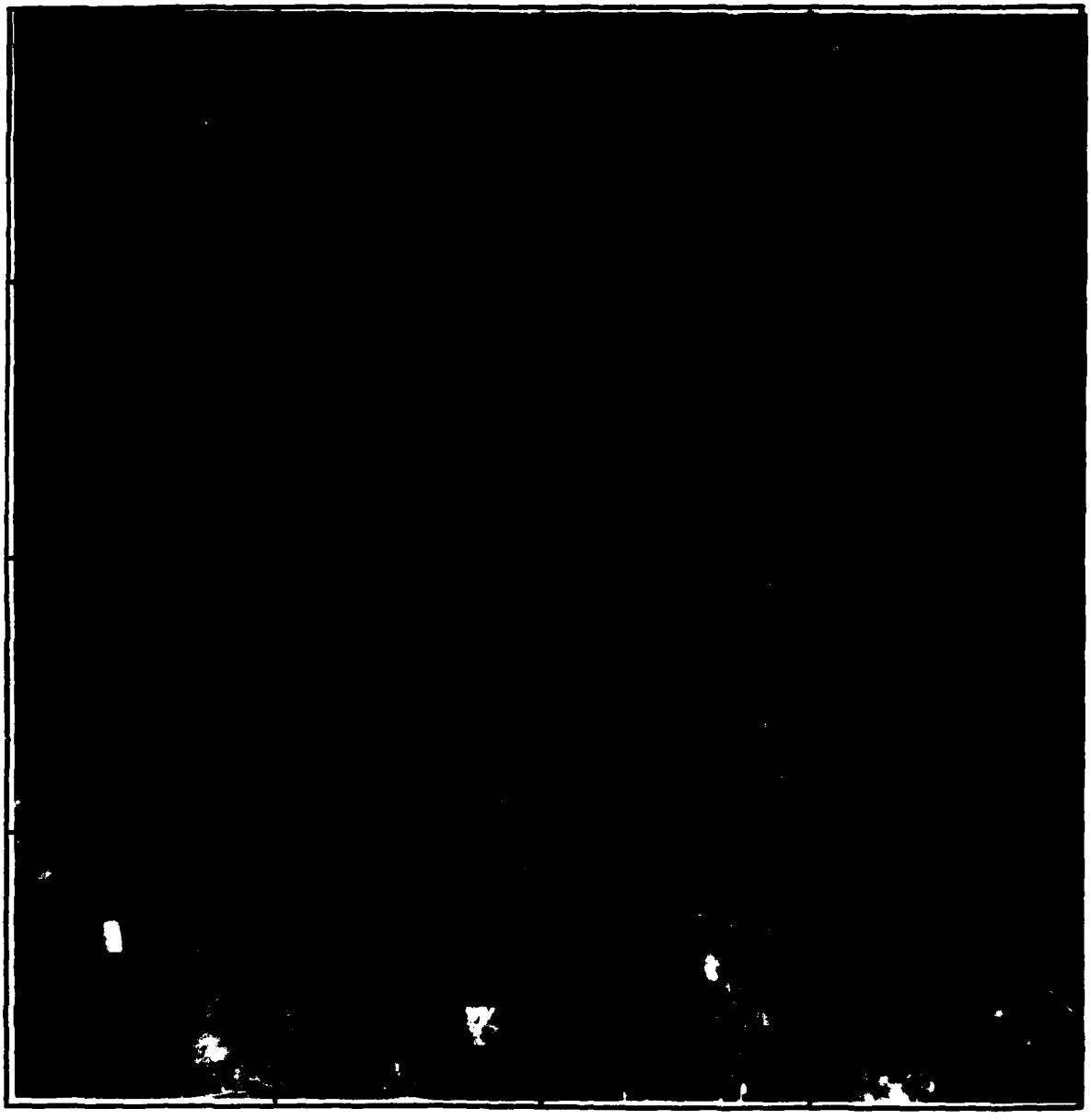


Fig. 4.15: 4000 x 4000 pixel section of image No. 54

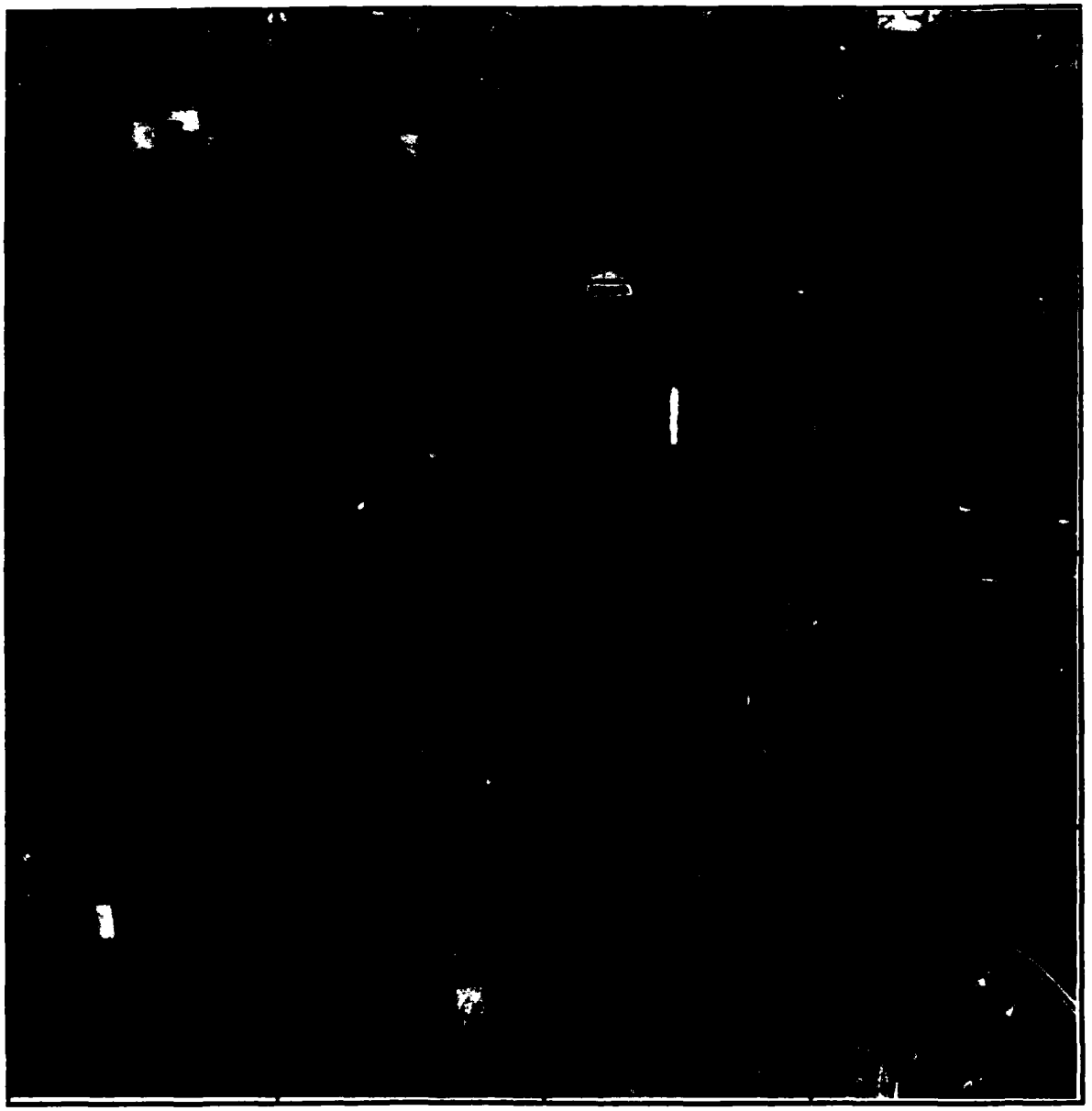


Fig. 4.16: 4000 x 4000 pixel section of image No. 56

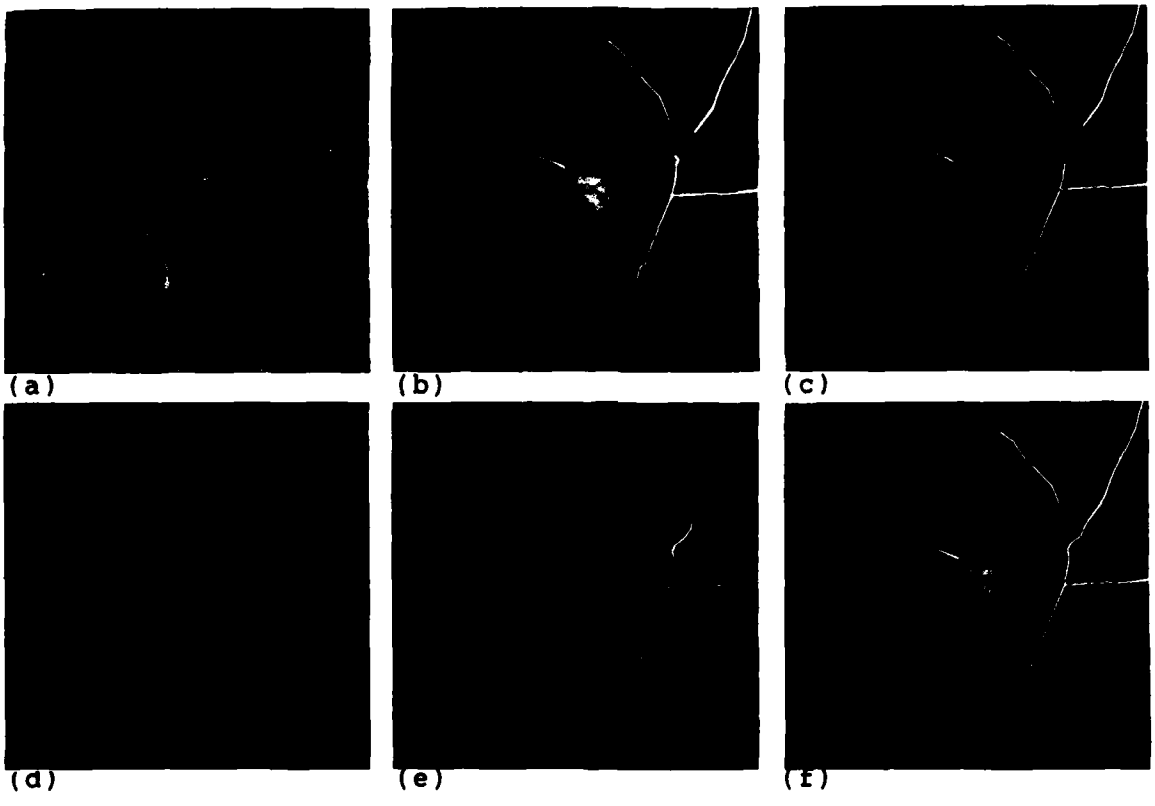


Fig. 4.17: Segment No. 2 of image No. 56

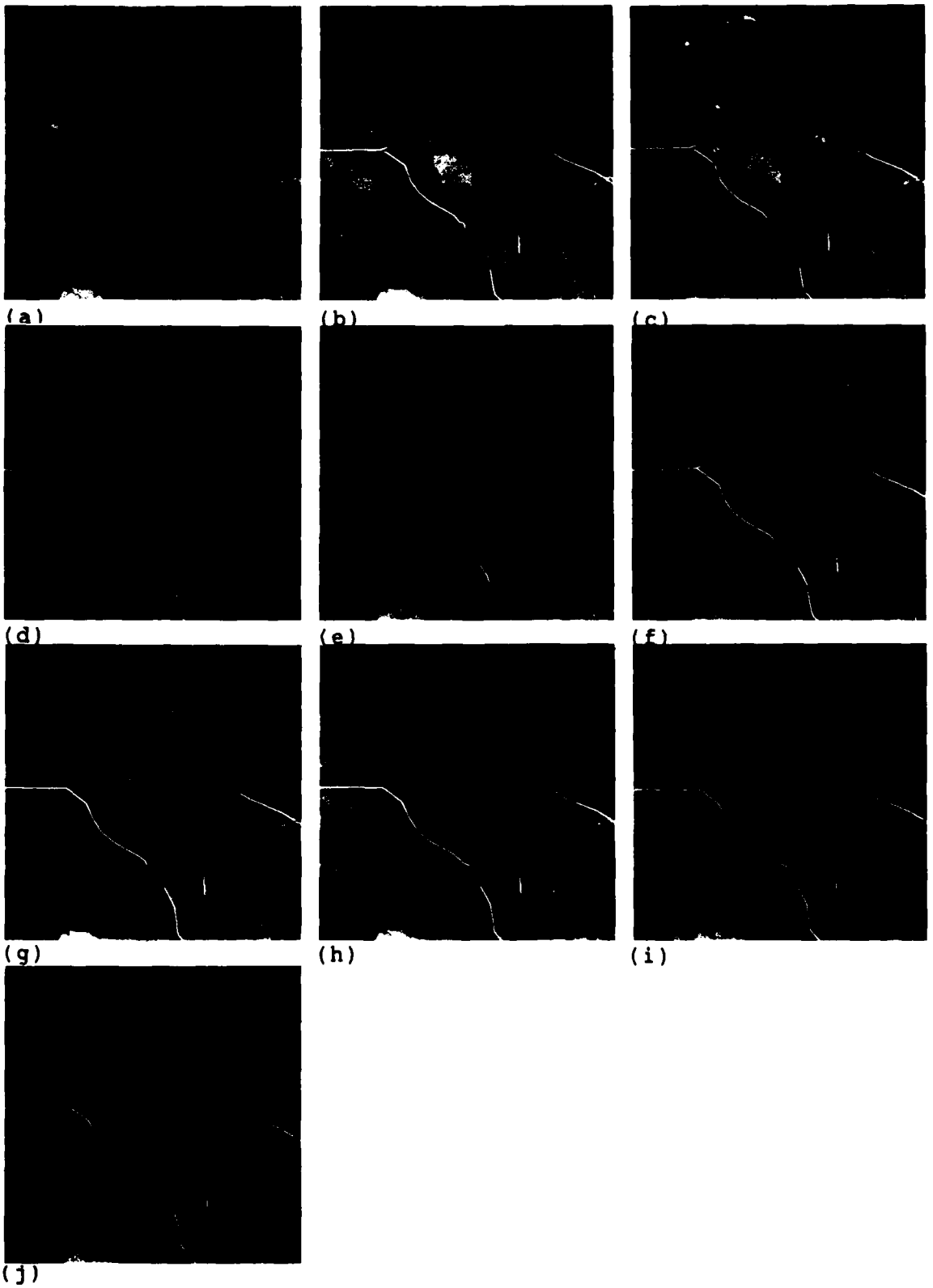


Fig. 4.18: Segment No. 3 of image No. 56

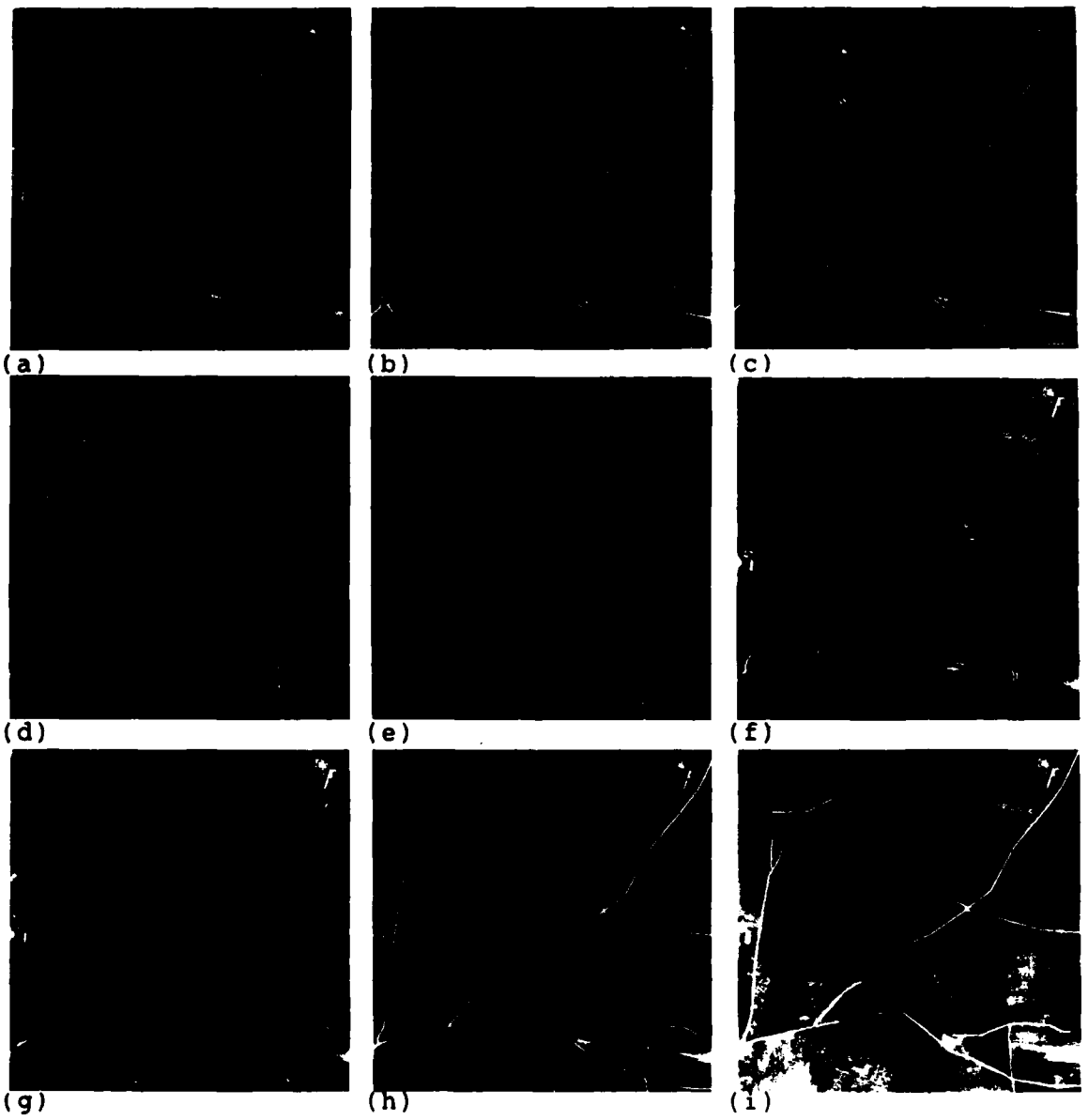
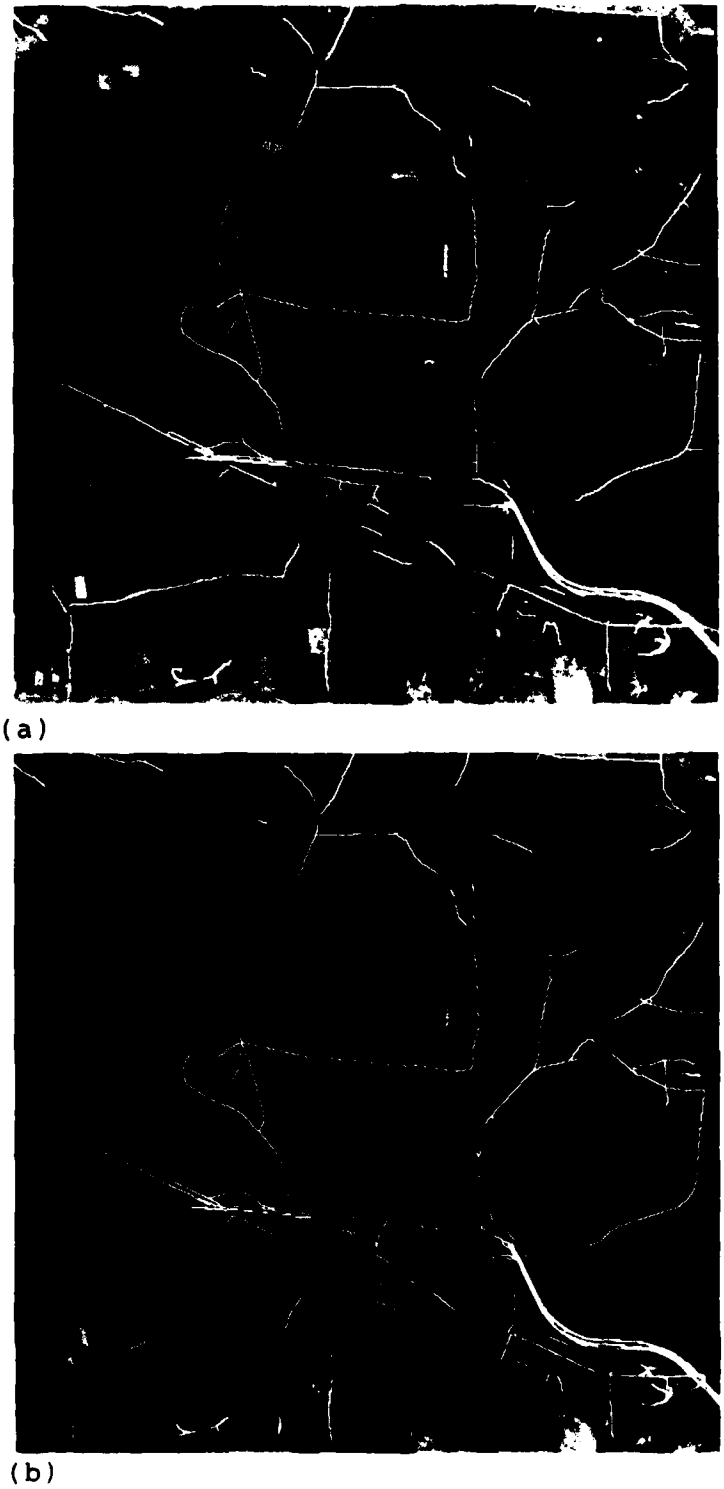


Fig. 4.19: Segment No. 8 of image No. 56

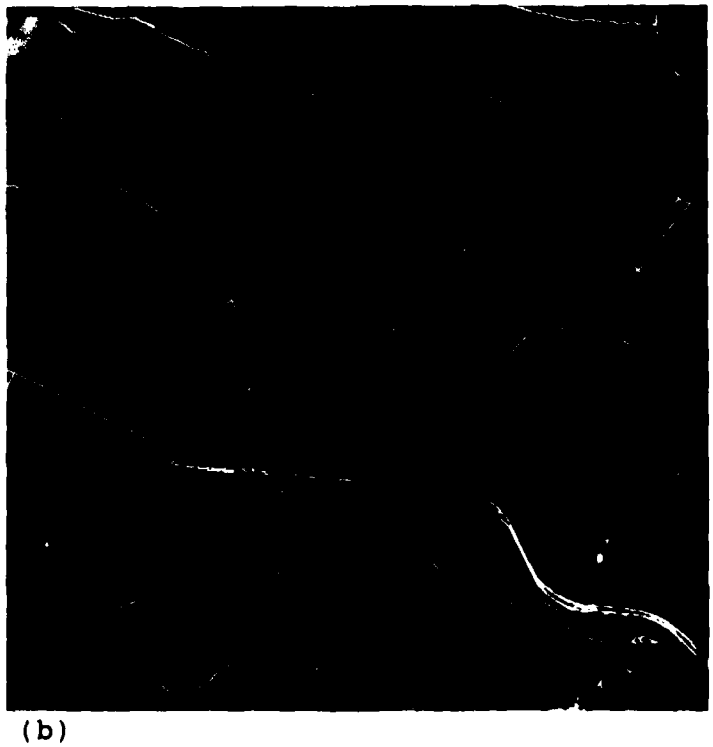


(a)

(b)

Fig. 4.20: Initial (a) and continued (b) extraction result of image No. 56

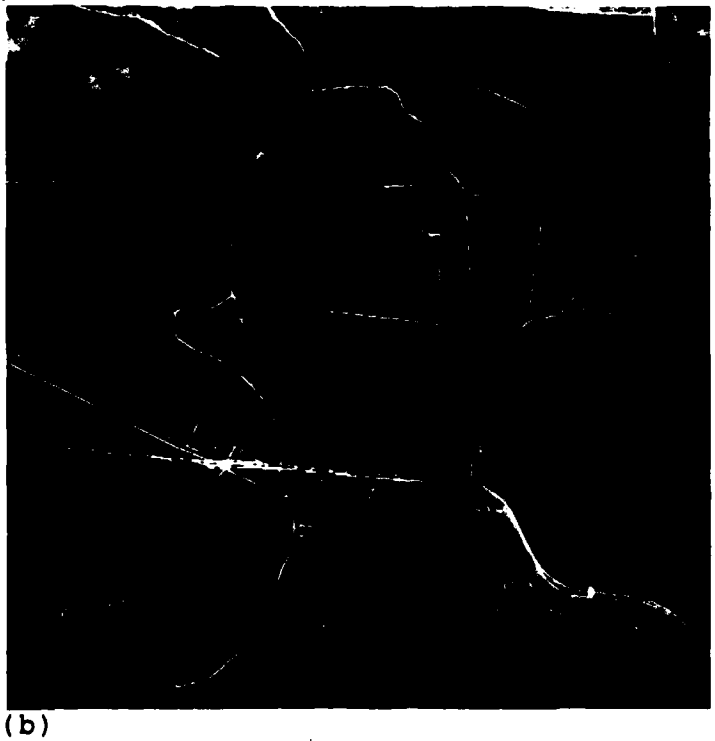
(a)



(b)

Fig. 4.21: Initial (a) and continued (b) extraction result of image No. 54

(a)



(b)

Fig. 4.22: Coincident (a) and continued (b) extraction result of image No. 56

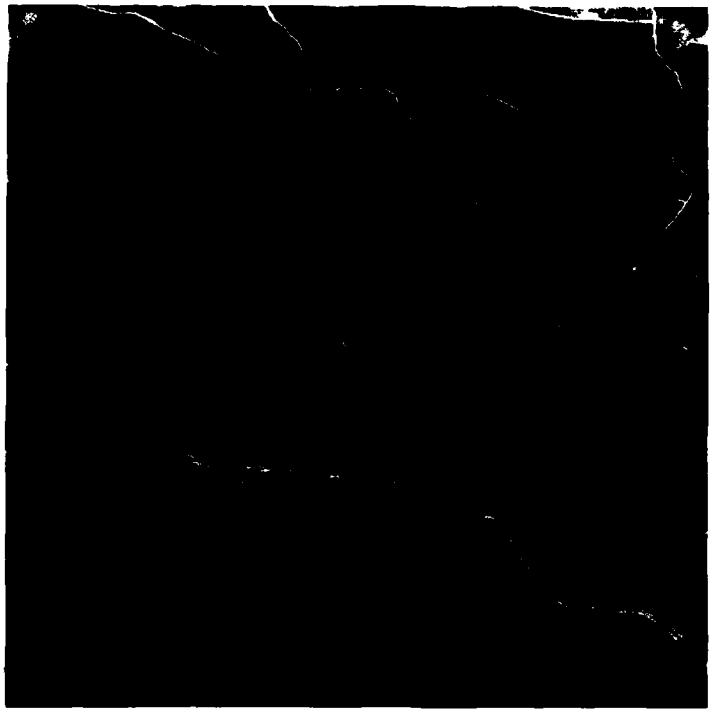


Fig. 4.23: Coincident extraction result of image No. 54

4.2.6 Discussion

The aim of the geometry-driven approach is to close those gaps in the extracted line network, which arise from the limitations of the local "view" of the old extraction procedures, when extracting line objects of poor appearance. To achieve this aim, the new approach makes use of some sort of global "view" by grouping the extracted line segments according to the geometric measures of distance and collinearity. This process improves the prediction for the extraction of missing line segments by the REM.

Our experiments with the new system show:

(1) The new approach leads to a completion of the extraction results.
(2) Certain constraints imposed on the REM by the standard set of parameter values can be reduced. Inspite of some more relaxed parameter values, the new extraction results are reliable too.
(3) A certain deterioriation in quality of the image data is tolerable. The set of parameter values was adjusted to the data of image No. 56. The same set of parameter values has lead to a reliable completion extraction from image No. 54, though the contrasts there are worse.
(4) The difficulties in extracting line segments corresponding to road segments in urban areas have not been reduced by the new approach. Usually, the situation in urban areas is too complex for an isolated signal processing approach. The use of semantics seems to be indispensible, in order to distinguish reliably streets and roads from parts of the surrounding objects.

4.3 The Image-Context-Driven Approach

The geometry-driven approach still leaves us with an incomplete extraction result at some places between paired CALCs. At some of these places the termination of the extraction process is definite, since there is nothing left to extract. At other places there seems to be still a chance for a continuation extraction, provided that some knowledge about the area objects between the CALCs is integrated into the verification process. Knowledge about the kind of objects, their shape and position in the AoI may motivate another attempt to extract missing line segments. To this end we have worked on the problem of image segmentation and object classification.

4.3.1 Raster Texture Evaluation

To study the effects of a region analysis by texture parameters, we chose different locations between paired CALCs where the road features were poor. At each such location we defined interactively an AoI. Two examples are displayed in figures 4.24a and b. They show AoIs from segment No. 3 and 4 of image No. 56. Different rasters of macropixels were superimposed on the AoIs. For each macropixel we calculated statistical texture parameters as defined by Haralick /3/. It was impossible to seperate differently structured objects by the parameter values of the respective macropixels. Either the macropixels proved too large or too small. In the first case they did not at all adjust to object boundarys. In the second case the samples were too small.

4.3.2 Interactively Supported Object Based Texture Evaluation

In order to confirm the suspicion, that the size of the samples and their conformity to the shape of the object are crucial for a successful texture evaluation, we switched to a different experiment. At the same locations as before, we defined by visual interpretation, the boundaries of the area objects. Closed contours were superimposed manually on the image data. Some contours are displayed in figures 4.25a and b referring to the examples of the

foregoing section . For each area object as a whole the same statistical texture parameters as before were calculated. The results were significantly better than in the first experiment. It was possible to distinguish differently structured objects from each other by their differing parameter values.

(a) (b)

Fig. 4.24: AoIs from segment No. 3 (a) and 4 (b) of image No. 56

(a) (b)

Fig. 4.25: Visually determined contours

4.3.3 Automatic Object Based Texture Evaluation

Obviously, the correct segmentation of the image data is mandatory for a successful texture feature classification. Therefore we turned our attention to the study of an automatic segmentation system developed at FIM. It would go beyond the scope of this report to discuss the system in detail. Therefore, we would like to refer the reader to a list of articles at the end of this report /4,5/. The system has been developed to work on multispectral

image data. Its application to monochromatic image data had to be studied and we will discuss it in the sequel from the viewpoint of this application. It has been designed to separate areas with different spectral features. A summary of its main functions is given in the following.

The first step of the segmentation process consists in the calculation of a contrast threshold for the generation of a binary image distinguishing regions of low contrast from regions of high contrast. If a clear distinction of these two types of regions in an image is possible, the histogram of the graylevel contrasts displays two maxima with a minimum in between seperating the set of all low contrasts from the set of all high contrasts. Thus, the contrast with minimum frequency is particularly suitable for the seperation of low contrast pixels from high contrast pixels. Since usually, the situation in aerial images is not as clear, more elaborate procedures are necessary for the calculation of the threshold. Still, the underlying idea is the same.

After the generation of the binary image, an extraction of compact, spectrally homogeneous centers, i. e. of centers containing only pixels of similiar grayvalues is performed. The process starts with the extraction of connected sets of elements in the binary image representing pixels with low contrast. The corresponding sets in the graylevel image are then subject to a shrinking algorithm, which reduces them to compact, spectrally homogeneous centers. During this process every pixel of a connected set is compared to all the other pixels of the set. If the number of graylevel contrasts greater than the contrast threshold is too big, the outer layer of the set is peeled off. This process is iterated until the number of contrasts greater than the contrast threshold is acceptable or the set becomes too small for the next processing steps.

The extraction of spectrally homogeneous centers is followed by the separation of classes of regions with similiar spectral features from the set of all spectrally homogeneous centers. A region is assigned to a class if all regions already assigned to

that class together with the region itself prove to be a spectrally homogeneous set of pixels. At the beginning every spectrally homogeneous center constitutes a class of its own.

Having defined classes of spectrally homogeneous regions, a classification process is started for all pixels of the image not yet classified, comprising all pixels dismissed during the region shrinking process and all pixels with high contrast. A pixel is assigned to a class, if the number of graylevel contrasts, which it generates with the pixels of that class and which are greater than the contrast threshold, is not too big.

Even after the last step there may be still unclassified pixels in the image, which do not belong to any already defined class due to their different spectral features. In order to get a complete segmentation result, these pixels are subject to a clustering algorithm. Clusters of pixels are accepted as new classes.

The clustering marks the end of the spectral analysis. The result consists of regions of different sizes, not necessarily compact, but of spectral homogeneity.

The preparation for the texture feature classification terminates with the extraction of spectrally homogeneous and spectrally inhomogeneous regions of a minimum size from the result of the spectral analysis. The spectrally inhomogeneous regions consist of several, adjacent, spectrally homogeneous regions of less than the minimum size. All regions are subject to split and merge procedures, which smooth their contours. The resulting regions constitute the input to the texture feature classifier.

Results of the whole process for the AoIs of segment No. 3 and 4 of image No. 56 are displayed in figures 4.26a and b. Different regions are represented by different grayvalues. On the whole, the segmentation results are not too bad, though some of the details cannot be confirmed visually.

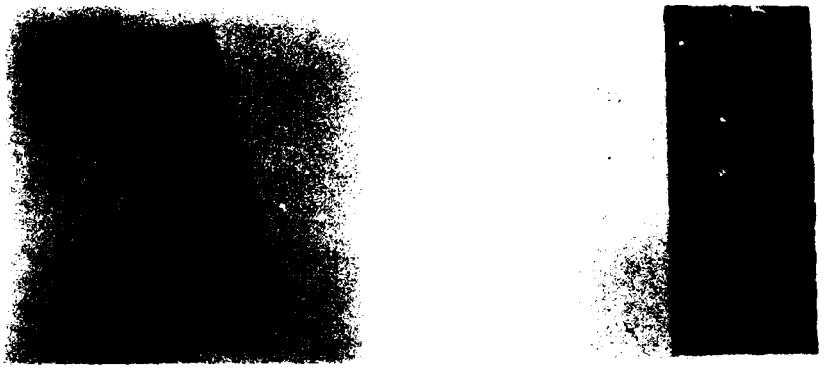
(a) (b)

Fig. 4.26: Automatic segmentation results

Texture parameters were calculated for the automatically defined regions and a clustering algorithm in the texture feature space was applied to the results. Figures 4.27a and b and figures 4.28a and b show the outcome for the sample AoIs of segment No. 3 and 4 of image No.56. They display the final segmentation results. Regions with similiar (dissimiliar) grayvalues are supposed to possess similiar (dissimiliar) texture features, figures 4.27a and b. In figures 4.28a and b the respective object boundaries are superimposed on the graylevel segments.

(a) (b)

Fig. 4.27: Texture feature classification



(a) (b)

Fig. 4.28: Automatically determined object boundaries

As can be seen from the two examples, the automatic object based texture evaluation cannot be judged as generally successful or generally unsuccessful in generating cues for missing road segments. The results for the data of segment No. 4 are satisfactory, figures 4.27b and 4.28b. On the other hand, the results for the data of segment No. 3 reveal a serious shortcoming, figures 4.27a and 4.28a. The distinction between the light forest and the adjacent fields in the upper part of the AoI is missing. Tests of the procedures at other locations of the image matrix also produced results of varying quality.

4.3.4 Assessment of this Approach

Summarizing the results of this investigation we come to the following assessment:

In agreement with other research results, we can state that a systematic raster texture evaluation does not produce a reliable basis for image segmentation and object classification when applied to aerial images of the kind as shown above. To improve the performance and automate the procedures, FIM had developed a combination of multispectral and textural feature evaluation. This software system had been adapted to and optimized for multispectral remote sensing input data. When we applied the system to our grey tone aerial test images, we found a degraded performance which did not fulfill our expectations. Obviously, the reason for the degradation is due to the lack of multichannel input data.

As our results show, we can improve the performance with interactive aids. But this was not our primary goal; we wanted the procedures to work automatically.

Although there seems to be optimization potential still, we conclude that

- with respect to the limited number of locations where a contextually driven image segmentation and object classification is necessary to support line object extraction

- with respect to the limited degree of automation and success possible for the application of the FIM multichannel image segmentation software system to grey tone aerial images

- with respect to the amount of computer time necessary for the generation of these segmentation and classification results

we cannot recommend the integration of (regional) context analysis via methods for multispectral and textural segmentation and classification into our procedure for line object extraction from grey tone aerial images.

5. Discussion of all Results and Conclusions

It could be shown that, compared to earlier results, the integration of some artificial intelligence aspects into our methods for line object extraction from aerial imagery produces line networks of a higher degree of completness and connectivity. The process still is fully automatic; some parameter values that depend on image type, size, and scale, can be changed interactively.

If the appropriate input data (i.e. a single aerial image or a stereo pair of images plus the relevant elevation data matrix) are provided, both the network of roads and the network of rivers and creeks can now be extracted automatically to a high degree of correctness, stability, and objectivity. Limitations exist for the following cases:

i if the contrast between the line and the surrounding areas disappears locally (e.g. a road between two fields of identical brightness or a river in a forest), the automatic extraction methods sometimes fail due to missing signals; the human interpreter still can interpolate these gaps from the basis of his "global" assessment of the situation. Sophisticated methods for automatic image interpretation would be necessary to solve this problem.

ii in densely populated urban regions the automatic extraction methods will not produce a complete and correct road extraction result due to the complex though mostly regular mixture of different objects. Again, intelligent, semantic based image interpretation would be required to solve this problem.

iii if noise, distortions or completely missing features prevail to an extend that even a human interpreter can suspect the true situation only vaguely, then we cannot expext automatic methods to produce correct line extraction results.

As it has already been mentioned, we did not pay the highest attention to the problms of optimization of computer memory and run time requirements; in this respect, optimization potential still exists for a refinement of our software system as well as for a special implementation on special purpose hardware.

Acknowledgement

The authors gratefully appreciate the valuable help of Dipl.-Inform. Karl Behrens, Dipl.-Ing. Edmond Mauer and Dipl.-Inform. R. Lubkowitz who contributed significantly to the completion of the project.

References

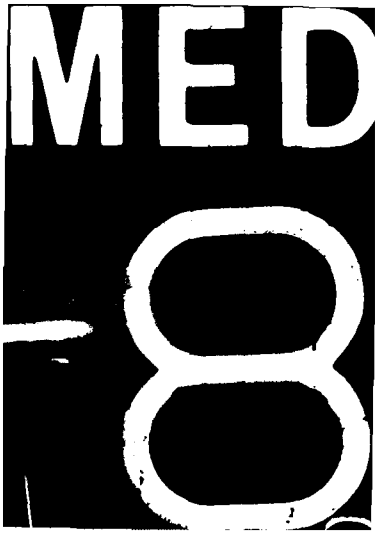
/1/ U. Bausch et al. — Automatic Extraction of Linear Features from Aerial Photographs Final Technical Report for ERO London, April 1981

/2/ H. Kazmierczak et al. — Coincident Extraction of Line Objects from Stereo Image Pairs Final Technical Report for ERO London, September 1983

/3/ Haralick, R.M. et al — Textural Features for Image Classification IEEE Trans. SMC, Vol. 3, No. 6 1973

/4/ Schärf, R. et al — Evaluation of Multispectral Images by Feature Combination 6-th Int. Conf. on Pattern Recognition of IEEE pp. 554-556, Munich, Oct. 1982

/5/ Mauer, E. — Aspects of Automation in a System for Remote Sensing Data Analysis by Feature Combination Proc. of 18. ERIM-Symposium on Remote Sensing of Environment pp. 981-987, Paris, Oct. 1984


MED
8